INTELLIGENT DECISION SUPPORT SYSTEM BASED ON HIERARCHICAL CAUSAL KNOWLEDGE STRUCTURES

Tetsuo SAWARAGI

KYOTO UNIVERSITY

1987

INTELLIGENT DECISION SUPPORT SYSTEM BASED ON HIERARCHICAL CAUSAL KNOWLEDGE STRUCTURES

Tetsuo SAWARAGI

KYOTO UNIVERSITY

1987

INTELLIGENT DECISION SUPPORT SYSTEM
BASED ON
HIERARCHICAL CAUSAL KNOWLEDGE STRUCTURES

by

Tetsuo SAWARAGI

dissertation submitted in partial fulfillment
of the requirements for the degree of

DOCTOR OF ENGINEERING

at

KYOTO UNIVERSITY

Kyoto, JAPAN

1987

PREFACE

The present studies have been carried out under the direction of Professor Sosuke Iwai in his laboratory at the Department of Precision Mechanics, Kyoto University. These are collected here as a thesis to be submitted to the Faculty of Engineering, Kyoto University, in fulfillment of requirements for the Degree of Doctor of Engineering.

Recent highly complex problems are difficult to deal with because of their inherently involving ill-structuredness, unboundedness and uncertainties. What is required of a computerized system that supports human decision making, a decision support system (DSS), is how to organize and integrate a great amount of domain-specific knowledge information so as to be utilized from various viewpoints of individual decisionmakers both adaptively and selectively. Here, the supports are assumed to be towards the decisionmaker's predecision stages, that is, a stage of problem finding, problem understanding and alternative formulation.

In this thesis, as a general architecture of such a decision support system, a knowledge-based approach is proposed and methodologies required to construct it are developed, where knowledge is a collection of causal relations among various problem elements that are described in experts' documents or reports.

The thesis can be divided into three parts.

The first definitional part of the thesis introduces the subject matter of the thesis and defines the boundaries of the topics included in the studies of decision support systems.

In the second part, knowledge is regarded as the one describing qualitative structural relationships among problem elements without considerations on problem substances. The methodologies for integrating the knowledge to quantify causal relationships are proposed graph-theoretically and knowledge retrieval system is presented on a basis of its structural properties. Using these quantities, individual decisionmaker's conformity for the alternatives with respect to the decisionmaker's psychological stress is measured, and a DSS is constructed that recommends the policy alternatives accordingly in an interactive manner.

In the third part, to explicate the conceptual and sophisticated principles implicitly represented in the surface knowledge, a methodology to organize hierarchical deep knowledge structures above the surface knowledge is developed. Based on this interwoven, memory-like knowledge structures, human cognitive processes, especially with respect to concept formation from instances as well as to hypotheses-driven prediction, are modelled. Then, this is applied to an interactive decision support system that can predict and interpret the events' occurrences by adaptively reorganizing and generalizing instances of episodic knowledge through their deep knowledge structure.

The author would like to express his sincere gratitude and appreciation to Prof. Sosuke Iwai for his directing the author's interest towards this research area and for his supervision and encouragement during the course of this thesis research.

The author is also deeply grateful to Associate Professor Osamu Katai of Kyoto University for his proper guidance and valuable discussions. The author's heartful thanks are also due to Dr. Kiyohiko Nakamura, a research associate of Tokyo Institute of Technology, for his useful comments at the early stage of this research while he was at Kyoto University.

Thanks are also extended to all members of the System Engineering

Laboratory, Kyoto University, for their helpful suggestions, discussions and collaborations.

Tetsuo SAWARAGI

Department of Precision Mechanics
Faculty of Engineering
Kyoto University
September, 1987.

CONTENTS

PART I.

GENERAL INTRODUCTION

The first part of the thesis is definitional. The two chapters introduce the subject matter of the thesis and define the boundaries of the topics included in the studies of decision support systems. With respect to this, conventional methodologies are briefly reviewed and basic concepts are introduced that are to be required in developing our suggesting methodologies in this thesis.

CHAPTER 1

BACKGROUNDS OF THE THESIS

1.1 Introduction

Recent highly complex societal problems involve the interrelationships between various aspects of the society such as economics, environment, and politics, where organizational decision problems such as strategic planning, policy formation, and marketing strategy are difficult to deal with primarily because they lack structure; they appear to be virtually unbounded and difficult to contain. The decision objectives and criteria may be only loosely specified, and the desires and preferences of the individual or organization are usually phrased in purely qualitative and ambiguous terms. Information characteristics for unstructured problems are summarized in Table 1.1 in contrast to the one for well-structured problems.

In facing decision problems like these, a decisionmaker must elicit, collect, sort, classify, evaluate, and structure sufficient information about the decision at hand to formulate rational alternatives in advance to the stage of choice. However, since there is likely to be confusion in the decisionmaker's mind at the outset concerning the nature of the problem domain, these stages of problem exploration and definition, or of alternatives formulation have been getting very difficult and complicated

Characteristics of Information	Ill-Structured	Well-Structured
Scope	Very wide Open	Narrow Closed
Level of aggregation	Aggregate	Detailed
Currency	Quite old (Historical)	Highly current
Required accuracy	Low	High
Type	Qualitative	Quantitative

Fig. 1.1 Information requirements

tasks requiring computer support as well as the stage of choice.

Decision support systems (DSS) represent information systems supporting unstructured decisions (Scott-Morton, 1971; Keen and Scott-Morton, 1978; Alter, 1980; Sprague and Carlson, 1982; Bennett, 1983). DSS are not to find the structure of the problem and automate it, rather to support a variety of unstructured decision process. The important implications required for the design and use of DSS are: (1) comprehensiveness (2) the user's flexible access to it, and (3) an extensive participation by the decisionmaker.

As the basis for describing the decisionmaking process, the well-known model by Herbert A. Simon has been proposed, which consists of three major phases: intelligence, design and choice, as shown in Fig.1.1 (Simon, 1960). The intelligence phase of the model includes activities to identify problem situations or opportunity situations requiring design and choice.

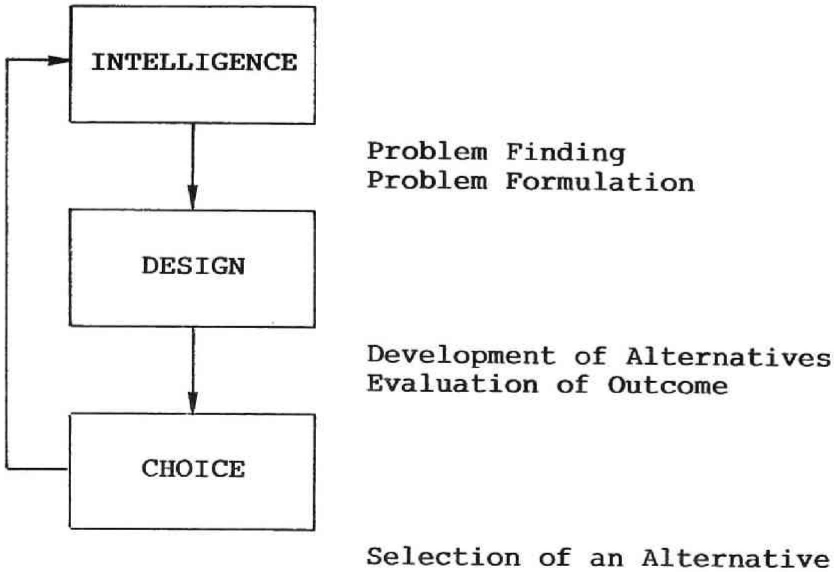


Table 1.1 Decision Process defined by Simon

Intelligence entails scanning the environment, either intermittently or continuously, depending on the situation. Problem finding is conceptually defined as finding a difference between some existing situation and some desired state. In succession to it, an aspect of problem formulation follows. The purpose of the problem formulation is to clarify the problem. Frequently, this requires some reduction of complexity, and it is often useful to establish an analogy or relationship to some previously solved problem or class of problems. The design phase includes the act of generating alternatives to be considered in the choice phase, which requires that there be adequate knowledge of the problem area and its boundaries (domain knowledge). Given these conditions, creativity can be enhanced by aids such as scenarios, analogies, brainstorming, template of decision process, etc. In this aspect, knowledge of outcome is important when there are multiple alternatives. The final choice phase is a selection of an alternative or course of action from those available, where a choice is made and implemented.

During World War II, many statistical strategies for helping decisionmakers make well-informed, rational, and consistent choices were introduced as actual applications of DSS under the name of operations research, and later under the name of management science. Since most of those strategies are totally objective and deterministic in the sense that they are largely mathematical-model based and quantitatively analytical algorithms, typically providing optimization or statistical inferences, they are most applicable to highly specialized clean-cut, and recurring problems, such as those involving systematic search, and the allocation of fixed resources.

The decision process which the traditional DSS methods aimed to assist was the last stage of human decision making process, that is, to choose the most rational and coherent one from the beforehand formulated set of alternatives under the definitely defined constraints. Wherein, decision makers were supposed to be able to distinguish clearly between what belongs to the set of alternatives and what does not, and numerical rankings of those alternatives are derived automatically. However, it is remarkable how little attention has been paid to how to support the intelligent and the design phases of decisionmaking process. To deal with the problems with respect to these phases, the *behavioral*, *descriptive* models are prefered rather than *normative*, *prescriptive* approaches mentioned above. The distinguishing characteristics of those models are summarized in Table 1.2. The descriptive model describes how decisionmaker actually make decisions and explains a variety of responses in human decisionmaking in terms of such a mental factor as *cognitive dissonance* and of *heuristics*,

Normative/Prescriptive	Behavioral/Descriptive
How people actually make decisions?	How people should make decisions?
Linear Programming Game Theory Statistical Decision Theory Utility Theory	Cognitive Dissonance Heuristics and Biases Human Information Process Decisional Conflict Psychological Stress
Maximization (Optimization) of Expected Utility	Bounded Rationality
Numerical Objective Function	Coping Patterns

Table 1.2 Normative vs. behavioral approches

which are closely related with his cognitive mechanism or capability of information processing (Steinbruner, 1974; Carroll and Payne, 1976; Slovic, Fishichhoff and Lichtenstein, 1977; Hogarth, Michand and Mery, 1980; Szaniawski, 1980; Tuggle, 1980; Einhorn ang Hogarth, 1981).

Cognitive dissonance, a theory by Leon Festinger, explains behavior after a choice is made (Festinger, 1957). An alternative that has some negative features, and rejected alternatives have some positive characteristics. A decisionmaker will tend to have cognitive dissonance, feelings of mental discomfort following a decision because of the recognition of negative and positive elements of the alternatives. A decisionmaker reduces cognitive dissonance by increasing the perceived difference in the attractiveness of the alternatives. This is done by avoiding information that might be contrary to the decision and by interpreting dissonant information in a biased way.

The use of heuristics in decisionmaking may be an efficient as well as effective method if the individual has adequate experience and judgment to make appropriate choices (Kahnemann and Tversky, 1972, 1973; Tversky and Kahnemann, 1973). A concept related to the use of heuristics is *bounded rationality* (Simon, 1977). Humans have a limited capacity of rational thinking; they generally construct a simplified model of the real situation in order to deal with it. Their behavior with respect to the simplified model may be rational. However, it does not follow that the rational solution for the simplified model is rational in the real situation.

The logical conclusion to be drawn from this formulation of the thought process is that some implicit conception or model of the flow of probable action in a situation must exist in the individual's mind before thinking can proceed. The implicit models of tasks, communication structures, appropriate behaviors, or possible outcomes of different kinds of situations serve as powerful organizing mechanisms in belief systems. These models need not be hopelessly complex; they simply need to take into account the patterns of activity and information that characterize the real political world for the political actor. That is, the slots in these learned models can be filled with the specific symbolism that identifies the actors, acts, means, ends, and scenic properties of the situation.

As for a model of such cognitive process, the pioneering investigations in the reconstruction on memory were conducted by Bartlett (Bartlett, 1932). Bartlett first observed that memories of even simple and vivid experiences tended to be incomplete. Moreover, different people exposed to the same input filled in the memory gaps differently in reproducing coherent scenarios of the past events. In short, Bartlett found that people actively reorganized their memories. He defined this memory organization unit as a *schema*. Schemata have been studied by a number of reasearchers in psychology and artificial intelligence (Kelley, 1972;

Axelrod, 1973; Bobrow and Norman, 1975; Minsky, 1975; Rumelhart, 1975; Bobrow and Winograd, 1977; Roberts and Goldstein, 1977; Anderson, 1980; Anderson, 1981; Anderson and Thorson, 1982; Embrey and Reason, 1986; Embrey, 1986).

Concerning with a similar idea to a schema, Abelson defines a *script* as a "coherent sequence of events expected by the individual, involving him either as a participant or as an observer" (Abelson, 1973, 1976, 1981). Scripts are the products of repeated experiences in particular situations that lead to general expectations about what happens in those kinds of situations. As relevant episodes are encountered in successive experiences with particular kinds of situations, they are incorporated into increasingly general and powerful scripts. Scripts may vary in complexity and content across different individuals. However, they tend to conform to the sequential organization of social process and share the key symbols of situations. Thus, scripts can be at once standardized and flexible. The more common and experimentally uniform the situations, the more standardized will be the scripts for those behavioral domains.

The schemata and scripts can explain well human information processing which consists of two elements of data-driven and conceptually driven psychological operations. Lindsay and Norman offer a preliminary distinction between the two types of operations by saying that conceptually driven processing " starts with the conceptualization of what might be present and then looks for confirming evidence, biasing the processing mechanisms to give the expected result." In contrast to conceptually driven operations which work from what is expected, a "data-driven system works from what is actually present" (Lindsay and Norman, 1977). That is, the interpretation is provided by casting the episode into a familiar cognitive (causal) schema which determines causal attribution. That is, people seldom

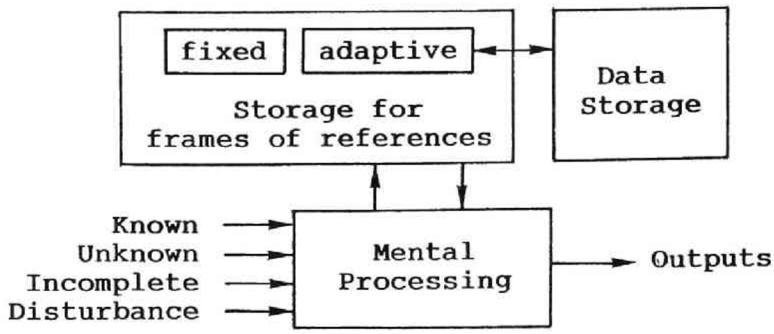


Fig. 1.2 Information processing model

respond exclusively to pure stimulus input. It is well known that any interpretive system that relies on prior experience will correct raw data according to statistical pattern in past experience. Moreover, it is clear that people respond to minor cues in social setting by imposing entire frames of references, scripts, or scenarios on the situation. These conceptual schemata then become a basis for understanding and responding to the situation and, perhaps, for shaping the eventual reality of the situation itself. Fig.1.2 shows the human information processing.

As for knowledge information processing, since 1970's, the research on applied artificial intelligence or *expert systems* has been getting active, which aims to utilize the experts' heuristical knowledge for solving such practical problems as various technological and medical diagnosis (Feigenbaum, 1977; Heyes-Roth, Waterman and Lenat, 1983). Expert systems can be considered as an instance of a decison support system. Concerning with the knowledge for the ill-defined problems, however, three fundamental problems get to be brought about, which have been missed to be sought in the traditional knowledge-based approaches in the expert systems (Sol, 1982; Sage, 1984; Methlie and Sprague, 1985).

At first, knowledge is to be clarified and presented to a decisionmaker in advance to its utilization in the further inferences. It is because a decisionmaker facing to such a complex problem cannot help feeling much procedural uncertainties, that is, about what knowledge, namely, information from experts, he should seek for, and about how to get the relevant knowledge from an immense amount of it both in efficient and comprehensive ways. Then, what is required to DSS is to help the decisionmaker focus his subject of interest in the vast amount of available knowledge, and to gradually or successively deepen his understanding about the problem, from global aspects to specific ones and from simplified views to detailed ones, and so on. Second, the knowledge concerning to that domain may consist not only of straightforward factual data but also of subjectively defined soft data such as experts' beliefs, judgement, opinion, and pure conjecture. Furthermore, such data, when they are elicited from different sources, are so often conflicting, which makes the scope of uncertainties much greater. Therefore, for DSS it is essential to have a knowledge representation formalism appropriate for such soft data and to make the decisionmaker understand those conflicts of opinions just as they are, rather than to resolve such conflicts in some black-boxed way. Moreover, such knowledge is by no means generally accepted one, but empirical, episodic knowledge on historical cases or particular instances with the lack of explicit generalization. The boundaries of required knowledge is not clear, and the decisionmaker's world is always changing and his decision takes place in a complex and partially unknown environment, in a word, an "open-ended" system. An architecture to deal with such an open system requires a kind of "self-organizing" capability to change its knowledge organization in response to changing conditions. The result is that the system automatically acts as if it has formed generalizations of particular instances, even though the system only stores individual instances. This requires of DSS to grasp the deep meaning implicitly implied in a collection of those episodic instances and to realize a capability of categorization and classification of its knowledge on a basis of its deep, semantic meaning. Finally, for an active participation of the decisionmaker, it is required of DSS to give him opportunities to estimate experts' knowledge and to integrate his own views or policies into that. In general, such soft knowledge offers different interpretations, and the knowledge-based DSS must be adaptable and extendable to meet the evolving goals of the user, that is, the decisionmaker. Therefore, it is required of DSS to contain any formal consideration of the subjective judgement and complex value structures that is an essential part of any important decision problem, which proves to be especially useful for assisting sound decisions in the face of uncertainties and conflicting opinions.

1.2 Objectives of the Thesis

On needs to develop a new architecture of DSS, in this thesis, we present a number of prototypical architectures and methodologies for intelligent knowledge-based DSS. The major objectives of the thesis are summarized as follows.

(1) Knowledge representation as Causal Networks

In decisionmaking, the documenting data written by experts are an important source of various expertises. Since the relationships described in documents on societal problems are mainly qualitative causal relationships between various elements of society, we propose a knowledge representation in computers as causally-chained network. Our formalism is proposed as to capture multiple perspectives including behavioral, cognitive, and organizational aspects the socio-political problems inherently have.

(2) An Integration of Causal Knowledge

It is required of DSS to integrate the causally represented knowledge into such a form that enables the decisionmaker to browse it and stimulates his intuitive judgment. As for this, we propose the integration methodologies from two different viewpoints: one is a method to quantify two kinds of causal effects carried by causal relations in a digraph-theoretical way. The other is a hierarchically chunking process, where the partial structures of a causal network are aggregated into larger conceptual units with different abstraction layers based on their semantic meaning.

(3) A Flexible Access to the Knowledge Base

Since the constructed knowledge base becomes so huge , it is difficult for a decisionmaker to grasp what is stored in the knowldge base in a comprehensive way or to focus his interest gradually deepening his understanding. Therefore, we develop a number of retrieval modes which enable a decisionmaker's easy access to the knowledge base, visual understanding of the knowledge structures, a readily shift of his interest or focus, and a summarized presentation at desired abstraction levels.

These are realized individually for two kinds of methodologies of knowledge integration described above.

(4) An User-Oriented Usage of the Knowledge

To meet the decisionmaker's own evolving goals or strategies in his usage of the knowledge, the items concerning with the mental burden enforced in his subjective judgment process should be taken into account. These are cognitive dissonance and cognitive heuristics. To reflect the slight differences of mental factors among individual decisionmakers on the system's reasoning outputs, two methodologies are developed: the one is the formulation of the decisionmaker's preference characteristics using numerical functions. The other one is the adaptive formation of the decisionmaker's mental models based on the methodologies for learning from examples, one track of researches on machine-learning in artificial intelligence.

(5) Modelling Cognitive Process of Human Intuitive Judgments

To construct intelligent DSS that can enhance and supplement human capabilities, it is required to clarify the mechanism of human cognition, especially with respect to his model-based or expectation-driven prediction and interpretations as well as to his category formation process from instances. We perform this modelling by introducing ideas from psychological domain: a schema (script) theory and prototype theory.

1.3 An Outline of the Thesis

The thesis consists of three parts; Part I, Part II and Part III.

Part I is a general introduction including this chapter and the subsequent Chapter 2, where some of the conventional decision theories and some of the basic concepts required in developing our suggesting DSS methodologies are briefly reviewed.

The main body of the thesis are divided into two parts, Part II and Part III, according to the difference of the treatment of the knowledge, i.e., the surface knowledge or the deep knowledge.

Part II describes the usage of causation knowledge that is encoded from the experts' documents without referring to any conceptual meaning implied in it. Basically, the causal knowledge is treated structurally with respect to its connectivities among nodes and signs carried by them on the basis of graph-theoretical considerations. Part II consists of three chapters, Chapter 3, Chapter 4 and Chapter 5.

In Chapter 3, a knowledge representation by cognitive maps is proposed, and procedures for encoding cognitive maps from documents as well as for constructing knowledge base by combining a number of maps are discussed.

In Chapter 4, at first, procedures to retrieve causal information from the knowledge base are discussed that enable the decisionmaker to form his overall and detailed understanding of the complex problems by watching it from various viewpoints. Then, a quantifying method for evaluating an universal relation, that is, a relation such that a cause element affects an effect element both in promoting and suppressing ways, is proposed as an integration of what is stored in the knowledge base.

In Chapter 5, the decisionmaker's preference characteristics on the quantified universal relation derived from the knowledge base are formulated with methodologies for identifying the decisionmaker's value function that reflects the psychological properties he feels in facing to the problem.

Part III deals with an usage of deep knowledge for decision support. Wherein, the key issue is not whether the sought information is eventually stored in the knowledge base or not, but how to support the decisionmaker by presenting more conceptual and sophisticated principles implicitly represented in those stored information. As a framework for the representation of such deep knowledge, a higher level knowledge structures of schemata and scripts are introduced and overlayed above the knowledge base. Moreover, an acquisition and an adaptive formation of these are discussed based on the ideas on category formation from the psychological domain. Part III consists of four chapters, Chapter 6, Chapter 7, Chapter 8 and Chapter 9.

In Chapter 6, a methodology for aggregating a detailed causal network into a more essential sequence of larger units is shown based on an idea of semantic primitives. A multi-layered knowledge structure with four different abstraction levels is organized, and this makes it clear how and when dynamically evolving patterns of interactions of themes, intentions and outcomes of social actions among participating actors come into play.

In Chapter 7, the hierarchical knowledge structure is organized as interwoven schemata, a frame-like structure, and stored in a schemata base in the system. In order to make an access to the knowledge smooth and flexible, the system is facilitated with a natural language interface that is compatible with the schemata structures. Human understanding process is discussed with respect to the ways of shifting focus in the hierarchical knowledge structure, and they are installed in the system as retrieval commands. The conversation with the knowledge base can be done through the front-end natural language interface and a variety of commands realizing the decisionmaker's global understanding of the situation or the context, as well as his detailed investigation into it.

In Chapter 8, a cognitive model is developed, which explains the process of human concept formation by being provided with instances as well as the resulting prediction process driven by the acquired concepts. Here, the characteristics of human intuitive judgments are discussed and modelled; to be anchored to the context or to the predefined concepts when he tries to judge newly encountering instances. The modelling is based on a prototype theory and is verified by being applied to a conceptual clustering of instances, where a measure of conceptual cohesiveness is introduced instead of conventional pair-wise similarity measures.

In Chapter 9, on the basis of what are discussed in Chapter 7 and Chapter 8, a procedure for adaptive formation of the decisionmaker's cognitive medels, called scripts, from the familiar episodes is described. This model is not a mathematically developed one but a roughly simplified conceptual model that makes the decisionmaker possible to recognize problems easier. Human heuristics in judgment in ill-structured problems are discussed with respect to this model, and an interactive DSS is constructed experimentally. The system can predict high-consequence events as well as interprete prior episodes based on accumulation of historical cases or episodes stored in its knowledge base.

Finally in Chapter 10, conclusions of the thesis are provided.

Table 1.3 summarizes the correspondences between the contents of the chapters and the five objectives of the thesis mentioned in the previous section in this chapter.

		(1)	(2)	(3)	(4)	(5)
Part I	Chapter 1	-	-	-	-	-
	Chapter 2	-	-	-	-	-
Part II	Chapter 3	X				
	Chapter 4		X	X		
	Chapter 5				X	
Part III	Chapter 6	X	(X)			
	Chapter 7		X	X		
	Chapter 8				(X)	X
	Chapter 9				X	X

Table 1.3 Outline of the thesis

CHAPTER 2

BASIC CONCEPTS FOR DECISION SUPPORT

2.1 Introduction

Decision analysis can be divided into four steps: structuring the problem; formulating inference and preference models; eliciting probabilities and utilities; and exploring the numerical model results. The traditional decision analytic research has all but ignored structuring, concentrating instead on questions of modeling and elicitation. The traditional approaches do exist for an overall analytic structure development, but are restricted to a few, usually hierarchical concepts and operations (Harary. Norman and Cartwright, 1965; Warfield, 1976; von Winterfeldt, 1980; Humphrey and Berkeley, 1983). Emphasis is put on simple, operational and computerized structuring. Little effort is spent on other phases such as a problem identification and a detailed content formalization, both of which by no means require any formal analytical way but domain-specific knowledge with respect to substantive problem aspects that delineate a problem in its real world context. Moreover, these phases are closely related with the human cognitive capabilities, and as a result, some kind of heuristic models concerning with the ways of human thinking should be introduced in the decision analysis.

In this chapter, the traditional analytical approaches are briefly reviewed mainly with respect to the decision tree analysis and the utility theory, as well as the hierarchical structuring researches such as ISM and AHP. Those methodologies' limitations are discussed and the necessities of the development of heuristic decision model are described for dealing with a problem identification phase. Then, the cognitive aspects in human decision behavior which are closely interelated with domain-specific contextual information will be discussed in detail. Finally, the schema theory from the field of psychology is presented, which is thought of as a useful framework for modeling human decisionmakers' heuristic models.

2.2 Ad Hoc Decisionmaking Methodologies

2.2.1 Analytical Approaches

Mathematics and probability theorists, starting as early as the 17 th centry, began to associate the concept of probability of hypotheses with credibility, "subjective" or "personal" probability emerged and has commonly been taken to represent degree of belief, although this view contains assumptions that are often inapplicable to everyday credibility judgments by humans.

Economics and statistical theories of preference and utility combine concepts of probability and desirability. Expected utility is a product of the probability of a statement and the desirability of its consequences. Probability is "objective" i.e., a countable relative frequency, a rate of m events to a total of n events. Again, to make the proper numerical assignments, the probabilities are assumed to add up to 1. Our view is that credibility is a concept that should be kept distinct from that of objective probability, in that objective probability requires an exhaustive and mutually exclusive set of alternatives to make numerical assignments and credibility does not. As human credibility functions in real life do not obey fundamental axioms of subjective or objective probability theory, there is good reason to give credibility a separate status (Colby, 1973).

Our society has, with increasing frequency, sought help from technical experts, trained in the application of formal analytical methods to problems of decisionmaking. The scientific approach originated during World War II from the need to solve strategic problems in situations where experience was costly or impossible to acquire. One of the offshoots of this early work has been the technique called "cost-benefit analysis," which attempts to quantify the expected gains and losses from some proposed action, usually in monetary terms. If the calculated gain from an act or project is positive, it is said that the benefits outweigh the costs and its acceptance is recommended, providing no other alternative affords a better cost-benefit ratio. Risk-benefit analysis is a special case of cost-benefit analysis in which explicit attention is given to assessing the probabilities of hazardous events and quantifying costs from loss of life or limb, pain, and anguish (Keeney & Raiffa, 1976).

The *decision tree* is a graphic aid within a larger context of the burgeoning science of decision analysis (Raiffa, 1968). The decision tree has at least three kinds of vertices. One is the *decision vertex*. Emanating from this vertex will be at least two edges. Each edge entry represents one possible consequence.

A small portion of a decision tree will illustrate these ideas. Figure 2.1 shows a single decision vertex labeled 1 from which several edges emanate. If the particular alternative represented by edge e_1 is selected, the edge leads to a *consequence vertex* labeled 2. From this vertex, several

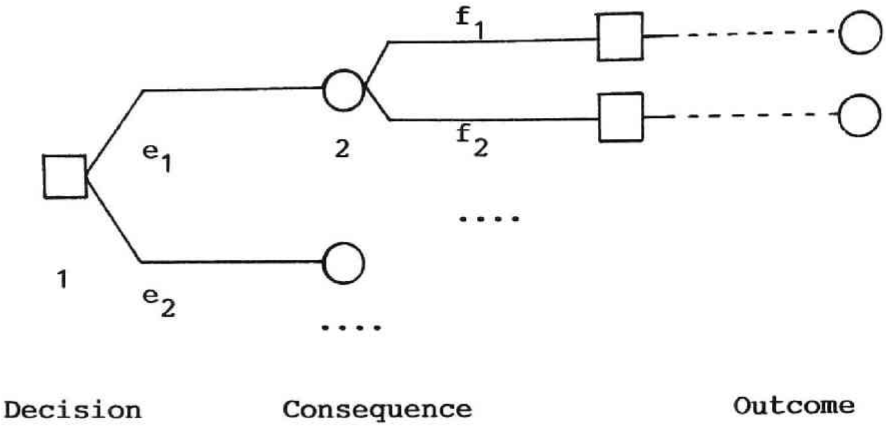


Fig. 2.1 Decision tree

edges f_1, f_2, ... emanate. Each edge represents a different possible consequence of choosing the alternative represented by edge e_1.

The entry relation accompanying branch e_1 is simply that there is a set of consequences that may ensue if alternative e_1 is chosen and this set is represented by the edges leaving vertex 2.

In using such a tree, probabilities will be assigned to each consequence, and costs or profits will usually be estimated for all branches of the tree.

The third type of vertices is an *outcome vertex*. It appears at the extreme end of a decision tree and has incoming edges but no outgoing edges.

A *decision path* representing a possible sequence of decisions and consequences leads to a particular outcome. By developing the various decisions and consequences thereof, with appropriate supporting data, the task of the decisionmaker is organized and made explicit.

The choice of one particular alternative for solving a problem in a complex system depends on value judgments. Those decisions depend heavily on the anticipated consequences of the alternatives considered. Such decisions are difficult for several major reasons, including multiple objectives, incommensurable evaluation factors, conflicting value systems, and uncertainty. A need exists for a systematic method of determining relative values of alternatives with multiple evaluation factors. Utility theory provides a basis for such a method.

Utility theory is concerned with the way in which people's preferences or judgments relating to value or worth are related to their decisions (von Neumann & Morgenstern, 1944, 1947; Fishburn, 1970). Utility theory can be classified in several ways: normative or descriptive, cardinal or ordinal, and probabilistic or nonprobabilistic.

Utility theory, like probability theory, originated in studies of gambling. Early theorists believed that the gamble with the highest expected winnings should be preferred and that wealth measured in cash is a utility function. This belief traditionally has been called "the principle of mathematical expectation." Utility theory is used in the proposed procedure to solve the large problem of evaluating complex alternatives by making many small evaluations and combining the results to obtain an overall evaluation which is consistent with the individual evaluations.

2.2.2 Structural Approaches

Structuring is an imaginative and creative process of translating an initially ill-defined problem into a set of well-defined elements, relations, and operations. The basic structuring activities are identifying or generating problem elements (e.g., events, values, actors, decision alternatives) and relating these elements by influence relations, inclusion relations, hierarchical ordering relations, etc. The structuring process seeks to formally represent the environmental (objective) parts of the decision problem and the decisionmaker's or experts' (subjective) views, opinions, and values. Graphs, maps, functional equations, matrices, trees, physical analogies, flow charts, and venn diagrams are all possible problem representations.

A generalization of the tree approach is *Interpretative Structural Modeling (ISM)* developed, for example, in Warfield (1974) and Sage (1977). In interpretative structural modeling, matrix and graph theory notions are used to formally represent a decision problem.

The methodology is being implemented in a man/machine interactive environment in such a way that human users are responsible for making subjective judgments while the computer is employed in an unobtrusive manner for bookkeeping and for performing and displaying the results of simple logical operations.

The mechanics of the interpretive structural modeling process are based on the one-to-one correspondence between a binary matrix and a graphical representation of a directed network. The fundamental concepts of the process are an "element set" and a "transitive relation." The element set is identified within some situational context, and the relation is selected as a possible statement of relationship among the elements in a manner that is also contexually significant. The elements correspond to the nodes on a network model, and the presence of the relation between any two elements is denoted by a directed line (or link) connecting those two elements (nodes). In the equivalent binary matrix representation, the elements form the contents of the index set for the rows and columns of the matrix, and the presence of the relation directed from element i to element j is indicated by placing a 1 in the corresponding intersection of row i and column j. In conducting an exercise, an individual or a group is subject to a series of queries of the form "Is element a related to b ?" or

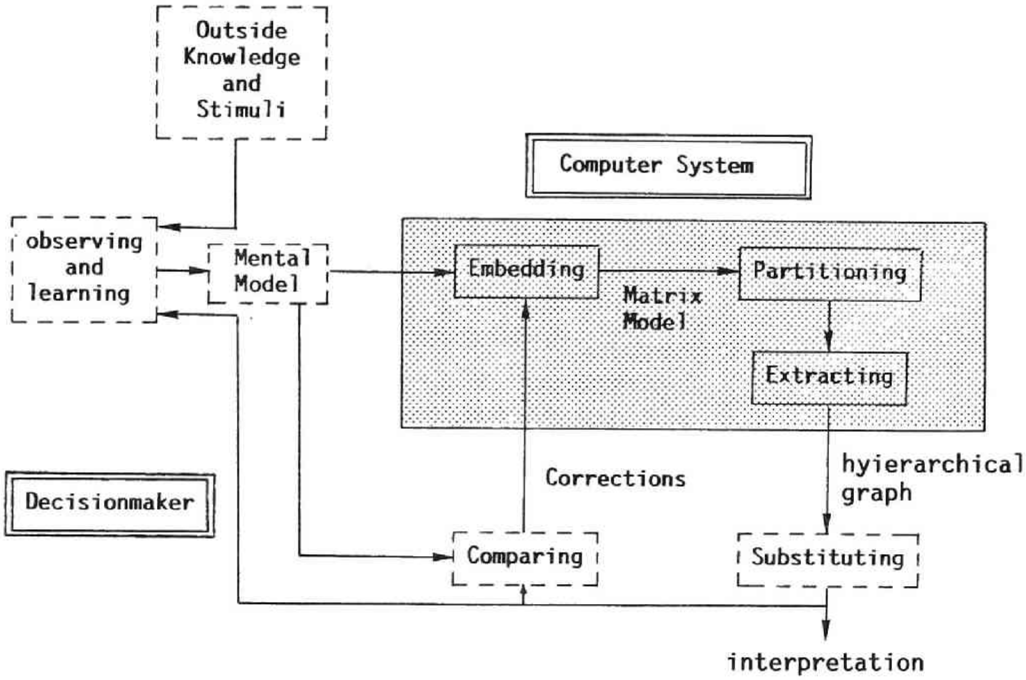


Fig. 2.2 Principal operations involved in ISM

a similar statement more appropriate to the context. The consensus view of the group as to whether a 1 or a 0, respectively, is entered in the appropriate location of the matrix.

Fig.2.2 is a representation of the principal operations involved in the methodology when implemented in a man/machine interactive mode. People are assumed to make observations in the real world and to draw upon their own knowledge and attitudes to identify pertinent concepts and relationships. The embedding operation is performed jointly by man and machine. The computer is supplied with an appropriate list of elements and the definition of a pertinent relation. A systematic sequence of queries is then generated, and a binary matrix representation of the system is assembled from the responses provided by a person or group of persons. When the matrix model is completed, computer operations are performed in order to partition the elements into natural hierarchical levels (partitioning) and to establish a minimal set of linkages which captures the entire pattern of the relation (extracting). The multi-level directed graph which results can be inspected and some symbols are introduced according to the context (substituting) to produce an interpretative structural model. If there are deficiencies or errors in the model (comparing), a correction process can be executed, again in partnership with the computer.

The methodology of ISM is widely applicable to the problem, and the definition of transitive relation between any pairs of elements determines the global structure organized from a viewpoint of that relation. Consider the nodes to be organized as the objectives in a given problem. No doubt the different objectives will vary widely in their scope, explicitness, and detail, and be inconsistent. Almost everyone who has seriously thought about the objectives in a complex problem has come up with some sort of hierarchy of those objectives. Fig.2.3 illustrates the overall objectives used by the environmental evaluation system (EES). The structure of EES is hierarchical in nature to account for the different levels of imformation for the impact analysis. Here, the methodology of ISM with the relation of specification of objectives will help the organization of hierarchy. Specification means subdividing an objective into lower-level objectives of more-detail, thus clarifying the intended meaning of the more general objective. Those lower-level objectives can also be thought of as the means to the end, the end being the higher-level objectives. Thus, by identifying the ends to very precise objectives (the means), we can build the hierarchy up to higher levels.

Another structuring methodology can be used here for the organized objective hierarchy. This is called *Analytic Hierarchical Process (AHP)* and was developed by T.L. Saaty (Saaty, 1980). He regarded the leaves in

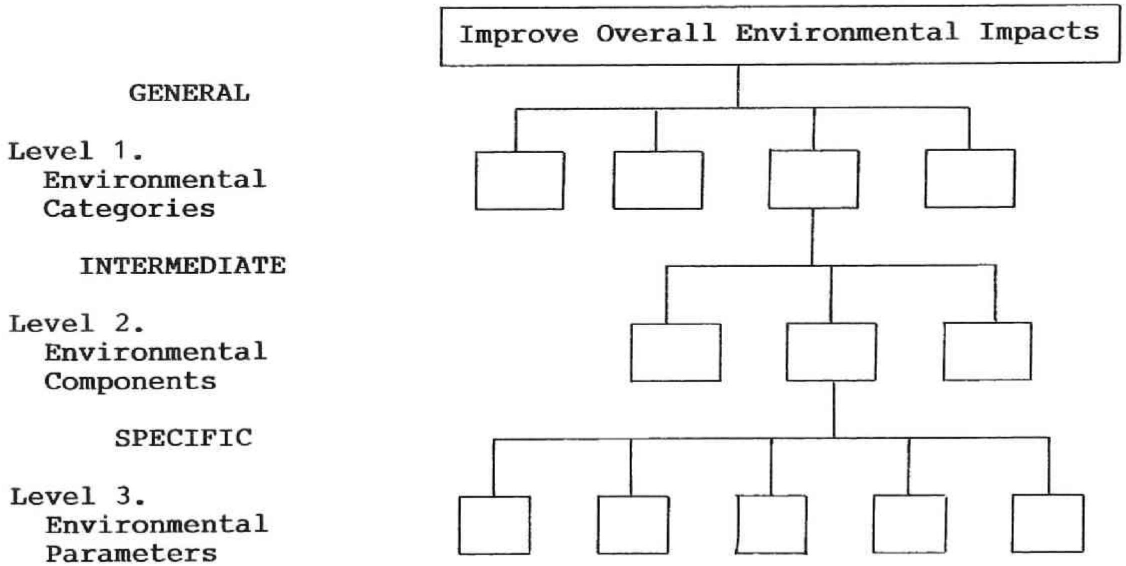


Fig. 2.3 Objective tree

the objective hierarchy, that is, the objectives that have no lower-level objectives, because the alternatives descending from the ends are assumed to be criteria for realizing the end, each of which can be divided into more precise criteria as proceeding deeply from the root in the hierarchy and getting to the leaves. The final purpose of AHP is to choose the best one from the set of alternatives by forcing a pairwise comparison of relative importance with some discrete weights among such a set of criteria (and alternatives) that has the common upper-level objectives.

Suppose there exists the objective tree as shown in Fig.2.4. At first, the decisionmaker compares criteria C_i and C_j for realizing the objective O with five or nine grades. That is, the computer program asks the decisionmaker a query in the form:

> "What is the relative weight of importance of criteria i against criteria j for the realization of the upper objective ?"

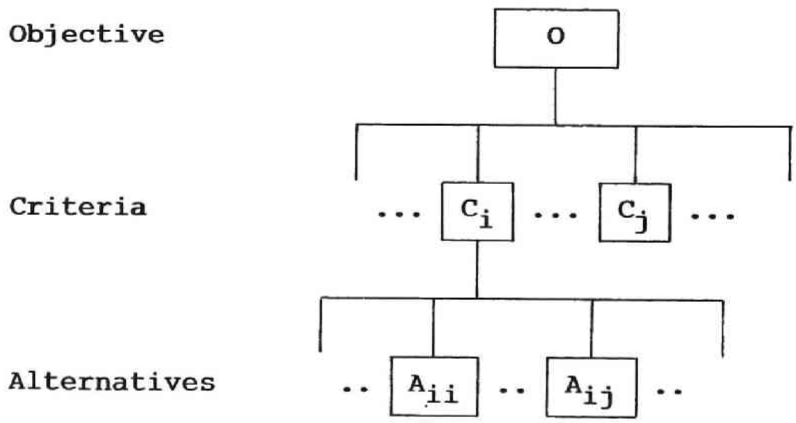


Fig. 2.4 Objective-Criteria-Alternatives hierarchy for AHP

From the results of the comparison, n x n matrix $\mathbf{A}=[a_{ij}]$ is formed, wherein a_{ii} and a_{ji} are assumed to be 1 and $1/a_{ji}$, respectively, so the comparison is required to be made n(n-1)/2 times for n criteria.

Let w_i be a weight for criteria i and $\mathbf{w}_c$ be a weight vector such that

$$\mathbf{w}_c^T = (w_1, w_2, \dots , w_n).$$

Since a_{ij} can be assumed to be equal to w_i/w_j,

$$\mathbf{A}\mathbf{w}_c = n(w_1, w_2, \dots, w_n)^T = n\mathbf{w}_c,$$

that is, $(\mathbf{A}-n\mathbf{I})\mathbf{w}_c = 0$, where $\mathbf{I}$ denotes a unit matrix. This equation denotes that an normalized eigenvector for the maximum eigenvalue equals $\mathbf{w}_c$. In the same way, the weight vector of alternatives for each criterion i, $\mathbf{w}_i^T = (w_1', w_2', \dots, w_k')$ can be calculated, where w_j' (j=1,..,k) is a weight for alternative A_{ij} with respect to criterion C_i. The total weight vector of alternatives $\mathbf{w}_t$ for the end can be obtained by

$$\mathbf{w}_t = \mathbf{w}_c \mathbf{w}_a.$$

Note that the depth of the objective hierarchy is not necessarily three-leveled but the procedures described above can be applicable in repetition to arbitrary-leveled hierarchy. In AHP methodology, as well as ISM, the deficiencies of pairwise comparison can be detected by calculating *consistency index (C.I.)* from the maximum eigenvalue of the formed matrix.

2.2.3 Common Problems of Ad Hoc Decisionmaking Methodologies

From the viewpoint of supporting a human decisionmaker in his realistic decisionmaking, the above mentioned methodologies are not sufficient in the following points:

(1) Lack of heuristic cognitive model

The methodologies are characterized by the fact that based on the human decisionmaker's series of partial judgments, the computer program makes up his global decision structure through some operations. For instance, in the utility theory the utility function for some attribute is derived by being interpolated from the judgments for the ideal or the worst case, and the global function is made up from the functions for each attribute according to the additive or the multiplicative rules under the condition of utility independence in a uniform manner. In ISM as well as in AHP, pairwise comparisons between any two elements leads to the decisionmaker's global decision structure based on the transitive rule, even though there exists a difference between them in that ISM deals with just binary comparisons, while AHP is extended to the relatively weighted comparisons. These operations are based on the prerequisite that the user, i.e., the decisionmaker, is always rational and consistent in his judgments, and as a result of that, the optimal solution can be obtained through analytic operations being provided with just a partial aspect of his decision structure. It may be true for the clear-cut, well-structured decision problems, where objectively defined evaluation criteria exist, but not in an ill-structured problem.

Simon defined a problem as ill-structured to the extent that it puts demands on the knowledge and repertory of problem-solving skills of the solver (Simon, 1973). What does a person do when he lacks reasonable confidence that the current explanation satisfies? To the extent that the task environment is relatively simple and unambiguous, a more satisfactory explanation is likely to be discovered with minimal problem-solving activity. How does the problem solver know that he has found a sufficient explanation? The person is not making a simultaneous choice among a set of possible explanations. Instead, he makes a series of choices among sets of promising components that are gradually built into a consistent, coherent pattern.

Ad hoc decisionmaking methodologies, however, have by no means any such heuristic model incorporated with respect to human realistic decision behavior. Indeed in those methodologies, the human decisionmaker does a critical role for the final result in answering queries made by the computer program, but he is dealt with as one of the components that does the task by being provided with some input and he presents some output. Wherein his cognitive behavior within that task is left a "black-box."

People ignore population base rates when making predictions about the category membership of a target member of the population, and do not employ base-rate information when called on to make predictions about the behavior of target cases. Even scientists are rather insensitive to sample size and are willing to draw reckless strong inferences about populations from knowledge about even a very small number of cases, which is expressed as "belief in the law of small numbers."

The early steps of such techniques, which involve structuring the problem and detailing alternatives and their attributes, may be useful devices for helping the decisionmaker think deeply and in relevant ways about his problem. However, the latter steps, involving quantification, may be forcing people to produce information at a level of precision that does not exist.

People view decisions based on shallow but nice-sounding rationales as better than decisions based on complex, thorough decision-analytic techniques. Decisions are made by searching for or constructing a good justification, one that minimizes lingering doubts and can be defended no matter what outcome occurs. We do have some hypotheses about factors that may not be viewed as adequate justifications. We think subjective factors, which are the cornerstones of sophisticated decision aids, such as tradeoff functions or probability judgments unsupported by frequentistic data, are perceived as weak justifications for decisions in the face of risk. Subjective probabilities, for example, leave one vulnerable to second guessing. Expected value computations may also make weak justifications because of their dependence on "long-run" estimates; such estimates may not appear relevant for decisions viewed as one-shot affairs.

(2) Avoidance of domain-specific contents

Several computer aids to support decisionmakers or experts in structuring decision problems typically rely on empty structuring concepts (decision trees, value trees, influence trees, or influence diagrams) and they guide the decisionmaker/expert in the analytic formulation of the problem. ISM and AHP are indicative of a trend in structuring research and perhaps in decision analysis as a whole. This trend turns the fundamentally empty structures of decision trees, goal trees, and inference trees into more operational, computerized elicitation tools, without adding problem substance. There are clear advantages to such an approach: a wide range of applicability, flexibility. user involvement, speed, limited training, and feedback, to name only a few. It also reduces the demands on the decision analyst's time.

As Warfield has observed, problems of systemic complexity are characterized by both content and structure (Warfield, 1974). Content involves the particular properties possessed by the elements of the system, e.g., the physical learning apparatus of the human is a system of elements such as the eyes, ears, and brain, which may be further described in terms of other properties. Structure is the configuration of relations among elements.

Recognition of the existence of both structure and content permits a division of labor in the analysis and synthesis of complex systems. In some cases, this division of labor may be only conceptual, i.e., one person has content knowledge of the properties of a system and also is able to structurally interrelate these elements. In other instances, one person or group may serve in the structural modeling capacity while another person or group concentrates on the content properties of the system.

There is, of course, the other extreme, the prestructured, prescanned problem specific version of decision analysis applicable to essentially identical situations. Such a structure and model can routinely be implemented with almost no additional training. In turn it gives up generalizability.

Neither extreme is totally satisfactory. Empty general structures must consider each problem from scratch. Substantive specific structures have limited generalizability. The middleground of problem driven but still generalizable structures and models needs to be filled. Problem taxonomies may help here by identifying generic classes of problems. But existing taxonomies are ill equipped for this task since they neglect substantive problem features. The question of filling in the middleground between 'too general' structures and 'too specific' structures thus becomes a question of searching for generalizable content feature of problems that identify generic classes of decisions. These generic classes can then be modeled and structured by "prototypical decision analytic structures" which are specific enough to transfer easily to other problems of the same class.

At the present stage of research this search process will necessarily be inductive because too little is known about problem substance to develop a problem driven taxonomy and matching analytic structures. An inductive research strategy may attempt to crystallize the generalizable features of a specific application, or simply use a phenomenological approach to delineate problem classes in a specific application area.

2.3 Decision Support Based on Heuristic Cognitive Model

2.3.1 DSS as an Interface System for a Human Decisionmaker

Because of the finite human information processing capabilities, all decisionmakers, if they are to cope , are forced to apply, often subconsciously, various simple strain-reducing cognitive, affective, and motor learning strategies for receiving, organizing, and processing the incoming information. Despite even the strongest motivation, a decisionmaker's limited memory, attention span, or decisionmaking capabilities may cause serious stress and strain, degrade reception, and produce inefficient learning.

In the normal course of business and life, there are many types of problematic learning situations that leaders, managers, and individuals face on a daily basis. Decision problems make up a significant portion of these, and they vary widely in their level of complexity and their degree of importance. Generally, decision problems can be divided into two broad classes: those that are important, and those that are not. The important decision problems are defined as those that are expensive to change in advising executives on how to survive and run a successful business.

To those who have dealt with such decisions in their own personal spheres, the enormous difficulties inherent in important decisionmaking are apparent. Moreover, the difficulties become greatly magnified in business and government decision context; the problems are usually far more important and complex, the scope of uncertainties much greater, and the risks and stakes of enormous magnitude.

Faced with an important decision, a decisionmaker must elicit, collect, assimilate, classify, evaluate, and structure sufficient information about the decision at hand to formulate rational alternatives and choose a logical and coherent course of action. The task is complicated, because the information bearing on the decision may consist not only of straightforward factual data but also of erroneous data and inherently soft data such as expert opinion, professional judgment, and pure conjecture. Furthermore, expert opinion and judgment, especially when they are elicited from different sources, are more often than not conflictive. To further complicate matters, most important decisions are also surrounded by future uncertainties that must somehow be taken into account. Finally, the criteria used by the decisionmaker to evaluate the alternative courses of action are usually conflictive and often poorly understood.

However, all of these strategies, such as linear programming and critical path analysis, are originally appropriate for objective and deterministic tasks; they ignore the human element; they omit any formal consideration of the subjective judgment and complex value structure that are an essential part of any important decision problem.

We must emphasize that in no sense does decision analysis replace decisionmakers with procedures devoid of human judgment. Rather, the purpose of the strategy is to provide an orderly, rigorous, easily understood, and intuitive-appearing structural framework, or standard, to aid the decisionmaker. The framework integrates the objective data surrounding a decision with the wisdom of experts and the decisionmaker's own personal judgment on the many topics that may have to be addressed to support the choice and implementation of a particular decision option. Thus decision analysis is strictly an accessory to decisionmaking: an organizing and clarifying framework that has proven to be especially useful for making sound decisions in the face of uncertainty, inconclusive evidence, conflicting opinion, and unclear personal judgments.

2.3.2 Cognitive Aspects of Realistic Actual Behavior

Both economics and psychology have recently converged on the notion that people base anticipations on their implicit causal models of reality. Thus, people do not necessarily extrapolate trends when making predictions; instead, they manipulate information in accordance with their models. In economics, this viewpoint is represented by proponents of the rational expectation hypothesis which postulates that the causal model is the "relevant economic theory." In psychology, the concept of *causal scheme* (or *schemata*) has become prominent through the influences of Kelly's *construct theory* (Kelly, 1972), Minsky's *frames* (Minsky, 1975), Abelson's *scripts* (Abelson, 1980) and several aspects of *attribution theory* (e.g., Heider, 1944, 1958). The idea, however, is not new. Bartlett introduced the notion of a causal schema to indicate the manner in which an individual represents in the mind the network of interconnections between events, objects, concepts or persons in any particular environment (Bartlett, 1932).

The need to investigate anticipations in real world settings should not be underestimated. Observation of real decisionmaking processes could lead to more different and more fruitful economic theories. In this point, we discuss four concerns arising from real decisionmaking activities:

(1) Procedural Uncertainty

Decision researchers have paid little attention to why people delay making decisions. Three types of nondecisions are refusal, delay and inattention. We consider that there are at least three sources of uncertainty: (a) lack of knowledge about events that could affect outcomes; (b) ambiguity concerning the consequences of actions with respect both to the specification of consequences and the individual's preferences; and (c) procedural uncertainty concerning means to handle and process the decision, e.g., specifying relevant uncertainties, what information to seek and where, how to invent alternatives and assess consequences, etc. Indeed, procedural uncertainty is excluded from formal models of decisionmaking. However, for unprogrammed decisions, our observations indicate that procedural uncertainty is both an important source of anxiety and can lead to decision delays.

(2) Concrete-Abstraction Continuum

The more he is burdened with many similar decisions to make, the more a decisionmaker is likely to employ hypothetical processes. The attainment of hypothetical processing mode on one decision task does not substantially generalize to another decision task. There is, in other words, repeated reversion to episodic mode on new decision tasks. It is instructive to speculate on the differences between concrete and abstract information and why concrete information has more impact.

(3) Information Overload

All seem to be acting rationally within the scope of their perceptions of the situation. It is often stated that 'information overload' (relative to human processing capacity) is a major obstacle in decisionmaking. Decisionmaking theories in both psychology and economics are well developed for structured decision problems. The important issues of real world decisionmaking, however, are much more concerned with developing a structure in the first place.

(4) Dynamic, Interactive Nature of Decisionmaking Processes

Considering decisionmaking over time adds a level of complexity in that the moment an actor makes a decision also becomes an important variable. However, as stated by Simon, decisionmaking needs to be studied across time since future anticipations are affected by past decisions (Simon, 1960).

2.4 Schema Theory for Modeling Cognitive Behavior

2.4.1 An Overview of a Schema

The term "schema" appears to have been first used by Bartlett in 1932 in his work on recall of perceptual experience (Bartlett, 1932). Since that time, it has been adopted as a descriptive or explanatory concept by so many researchers in social psychology (Norman & Bobrow, 1975; Palmer, 1975; Rumelhart & Ortony, 1977; Rumelhart; 1980) and artificial intelligence (Bobrow and Norman, 1975; Minsky, 1975; Bobrow and Winograd, 1977; Roberts and Goldstein, 1977; Embrey and Reason, 1986; Embrey, 1986).

The processing of information involves scanning the environment, selecting items to attend to, taking in information about those items, and either storing it in some form, so that it can be retrieved later for consideration, or using it in some form, so that it can be retrieved later for consideration, or using it as a basis for action. It goes without saying that there is necessarily a tremendous amount of selectivity in this process, because we cannot notice every detail in the environment. To select the information that is useful and to process it quickly and efficiently, the perceiver needs selection criteria and guidelines. Hypothesis-driven processing is processing that is guided by expectations or preconceptions. The hypotheses, whatever their nature, tell the social perceiver what data to look for and how to interprete the data that is found. Hypotheses lend focus, structure, and sequence to search behavior and subsequently provide a basis for how the information will be used. Neisser has called this kind of processing schematic processing (Neisser, 1976).

A schema is a cognitive structure that consists in part of the representation of some defined stimulus domain. The schema contains general knowledge about the domain, including a specification of the relationships among its attributes, as well as specific examples or instances of the stimulus domain. As such, one of the chief functions of a schema is to provide an answer to the question. The schema provides hypotheses about incoming stimuli, which include plans for interpreting and gathering schema-related information. It may also provide a basis for activating actual behavior sequences or expectations of specific behavior sequences for how an individual behaves in a social situation.

A schema can be thought of as a pyramidal structure, hierarchically organized with more abstract or general information at the top and categories of more specific information nested within the general categories. The lowest level in the hierarchy consists of specific examples or instances of the schema. The schema is connected to other schemata through a rich web of associations among the schemata, indicating these cross references. In addition, one can have schemata for the lower level specific instances of another schema. Fig.2.5 schematically illustrates the schema organization.

```
Generic Restaurant Schema

   Is_A:              Business_Establishment

   Types:
         range:       (Cafeteria, Seat_Yourself, Wait_To_be_Seated)
         default:     Wait_To_Be Seated
         if_needed:   IF wait_for_waitress_sign or reservation_made
                                then Wait_To_Be_Seated,

                      ...............

   Location:
         range:  ,    an ADDRESS
         if_needed:   (Look at the MUNU)

   Name:
         if_needed:   (Lool at the MENU)

   Food_Style:
         range:       (Burgers, Chinese, American, Seafood, French)
         default:     American
         if_added:    (Update Alternatives of Restaurant)

   Times_of_Operation:
         ................

   Payment_Form:
         ................

   Event_Sequence:
         default:     Eat_at_Restaurant Script

   Alternatives:
         ................
```

Fig. 2.5 Illustration of schema organization
— "restaurant schema"

2.4.2 Schema Functions to Overcome the Limitations of Human Information Processing Capabilities

Given some stimulus in the environment, it would be prohibitively time-consuming to match it against all prior experiences and given some problem to solve, the problem would never get solved quickly enough if one were to try out every possible solution. Even if the stimulus configuration were identified quickly and a strategy for solving a problem were selected immediately, it would still be prohibitively costly to process data piece by piece instead of in larger chunks. What schemata do is to enable the perceiver to identify stimuli quickly, and select a strategy for obtaining further information, solving a problem, or reaching a goal. The functions of schemata can be summarized into the following five points:

(1) The encoding and representation functions of schemata

When a stimulus configuration is matched against a scheme, elements in the configuration come to be ordered in a manner that reflects the structure of the scheme. The basic structure imputed to the configuration is a hierarchical one based on levels of abstraction. In addition, schemata enable the perceiver to group classes of observations together and to order them in particular ways. The structuring function of schemata is important because it influences subsequent recall of information and provides the basis for inferences and predictions.

(2) The functions of schemata on easiness of recall from memory

Either imposing a scheme on a stimulus configuration or encountering a stimulus configuration that is a good match to a schema increases overall recall, especially recall of schema-relevant material. Under some circumstances, moderately inconsistent information will be recalled better than schema-consistent information. The effects of schemata on recall seem to be due to both encoding and retrieval biases, although encoding biases seem to be stronger.

(3) Supplement of missing data

This process can occur in several ways. First, the schema can direct a search for data. That is, the schema contains characteristic attributes and dimensions associated with a particular stimulus domain. When a configuration is matched to a schema and the configuration is missing certain attributes usually found in that domain, search behavior may proceed for those attributes so as to obtain a fuller match. Schema also fill in missing data in a second, more interesting way. When data along some dimension or attribute is missing, the schema may fill in the missing values with best guesses regarding what the value should be and insert this value in the configuration. Minsky terms these best guesses *default options* (Minsky, 1975). These default options presumably develop from experience with instances of the schema and thus are typical qualities of the stimulus domain in question. Default options may not only be inserted when a particular piece of data is missing; they may also be inserted in the stimulus configuration if there is too much information for the perceiver to process.

(4) Schemata as a basis for evaluating experience

A schema represents a normative structure, and as such, specific instances can be matched against it for goodness of fit. There are several implications of this point. One is that if schema-relevant expectations are generated prior to encountering any given stimulus configuration, the configuration is judged against these expectations. Expectations, in turn, exert a powerful influence on emotions and behavior. Another implication of the normative function of schemata is that people who have highly developed schemata will make more extreme evaluations for schema-relevant material than will people whose schemata are less well developed.

(5) Schemata as a basis for anticipating the future, setting goals, making plans, and developing behavioral routines

Schemata enable the perceiver to set goals and enact the appropriate behavior sequences. Having structured representations of events, persons, roles, or relationships enables a social perceiver to envision a state toward which he or she would like to move. Schematic conceptions of how the world works can also help create the reality they anticipate, even in the absence of objective bases for it in the environment. Schemata enable the perceiver to predict the future by specifying what events, abilities, or behaviors have a high likelihood of occurrence when activated; they also prompt the perceiver to see schema-relevant events, attributes, or behaviors as more likely to occur, even when the schema is subsequently discredited. Under some circumstances, the knowledge portion of the schema is tied to guidelines for actions, thus enabling the individual to act in the service of relevant goals. When this schema is strongly enough evoked and schemata-relevant behaviors are set in motion, the implications of the schema may be strong enough to actually create the reality that is implied.

2.4.3 Schema and Decision Behavior

The recent focus of artificial intelligence is on ways to program computers to "understand" natural language texts and visual scenes by using higher order knowledge structures that embody expectations guiding lower order processing of the stimulus complex. To summarize these developments, artificial intelligence workers concerned with statements about real-world events find the structure called *scripts* seen in relation to familiar

```
EAT_AT_RESTAURANT Script (EAR)

    Props:               (Restaurant, Money, Food, Menu, Tables, Chairs)

    Roles:               (Hungry_Persons, Wait_Persons, Chef_Persons)

    Point_of_Views:       Hungry_Persons

    Time_of_Occurrence: (Times_of_Operation of Restaurant)

    Place_of_Occurrence:(Location of Restaurant)

    Event_Sequence:

       first:    Enter_Restrant Script (ER)

       then:     if (Wait_To_Be_Seated_Sign or Reservations)
                    then Get_Maitre_d's_Attention Script (GMA)

       then:     Please_Be_Seated Script (PBS)

       then:     Order_Food_Script (OFS)

       then:     Eat_Food_Script (EF) unless (Long_Wait) when
                    Exit_Restaurant_Angry Script (ERA)

       then:     if (Food_Quality was better than Palatable)
                    then Compliment_To_The_Chef Script (CTTC)

       then:     Pay_For_It Script (PFI)

       finally: Leave_Restaurant Script (LR)
```

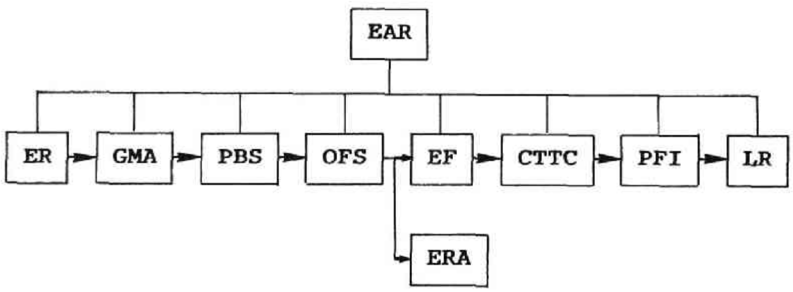


Fig. 2.6 Illustration of script

contexts rather than in isolation (Abelson, 1973; Stevens, Collins and Beranek, 1977; Schank & Abelson, 1977; Stevens, Collins and Goldin, 1979; Carbonell, 1980, 1981; Schank and Riesbeck, 1981). Fig.2.6 shows Schank & Abelson's "restaurant script." This structure produces great savings in computational efficiency and in apparent "understanding." Understanding a situation can be taken to mean the cognitive retrieval of previous situations to which the present situation is similar. Appropriate behavior in the present situation is then specified by the behavior in those previous situations. This formulation is most clear for situations which cover many repetitions which have built up an easily identified equivalence class, that is, scripts.

It is instructive to consider a script as a cognitive category and to consider various episodic instances of its performance as members of the category. Category theory has recently undergone a shift of orientation within cognitive psychology and philosophy. The classical view that categories of objects are defined by sets of necessary and sufficient features has been challenged. Even if taxonomies could agree on the unique defining features of particular objects, it seems psychologically implausible that there exist well-articulated, all-or-nothing boundaries for common categories. The newer theoretical view is that a category is defined not by absolutely criterial attributes, but by *family resemblances* among its members, just as all the siblings in a family can look somewhat similar without any single feature being common to all (Rosch, 1973, 1978; Rosch et al., 1976; Rosch and Mervis, 1975). Experimental subjects do not necessarily agree whether any object is definitely in a category, but there is high reliability from groups of subjects who rate relative degree of in-ness or out-ness. This judgment refers to the so-called *prototypicality* of objects. Rosch and Mervis have shown that to the extent an object's attributes are statistically common among other objects in the category, the object is judged more prototypical of the category (Rosch & Mervis, 1975).

What we call *tracks* of scripts are the those that contain many low-level particulars about events, which are slightly different from the particular for other tracks. At the extreme are abstractly stated scripts with a minimum of specification and are called *metascripts*. The defining feature of a metascript is that many of its scenes are specified at a higher level of abstraction than are those of a script. These abstract scenes subsume different concrete realizations. Memories for script episodes are not necessarily stored with the presentation for the particular script. Rather, memories may be stored at a metascript level, promoting confusion as to which concrete script was involved.

Just as two or more objects are similar to the extent that they have many shared features but not many unique features, so two or more episodes are similar to the extent they have many shared events but not many unique events. Script episodes have this critical similarity property: Separate experiences of the same script invoke a common core of events and scenes with relatively few variations. Scripts have internal coherence. Furthermore, scripts have potential similarity properties in relation to other scripts. Any script can in principle be compared with any other script for how many of their prototypical scenes overlap and for how many are unique. This resulting measure is called external similarity.

When a proposed script, however internally coherent its episodes, has high external similarity, it is not a very efficient knowledge structure. By contrast, some scripts have not only internal coherence but also external distinctiveness. The general rule for efficiency would seem to be the maximization of the ratio of internal coherence to external similarity. This principle applies at all levels of abstraction. If there is too much external similarity at a given level, then similar structures are grouped together to increase coherence and redefine what is external.

2.5 Concluding Remarks

In this chapter, we put emphasis on the importance of the computerized support for pre-decision stages preceding the selection. That stage includes understanding of the problem itself and structuring it so that the kinds of decision analytic methodologies can be applied to it. Since this stage has been an art left to the intuition and craftsmanship of the individual analyst and requires the deep understanding of the problem domain, the computerized support should be in an architecture of knowledge-based style and should contain the human decisionmaker's heuristic model within it.

First, the traditional decision analytic approaches as well as some of the representative structuring researches so far were briefly reviewed, and their insufficiencies were discussed from a viewpoint of realistic decision behavior in an ill-structured situation.

Next, in contrast to the rationalized, optimality-seeking decisions sought by those traditional methodologies, psychological decision behavioralism was surveyed. That idea denies the existence of the rationality and of the optimality under the ill-structured situation, and even though these existed, they could not be realized because of the limitations of the human information processing capability. Instead, the analyst overcomes that limitation by adopting heuristic models that are closely related to a memory organization as an accumulation of individual experiences.

Finally and with respect to that, the schemata theory was reviewed, which is thought to offer a conceptual foundation for realizing such heuristic models in constructing an intelligent decision support system for ill-structured problems.

PART II.

DSS BASED ON SURFACE KNOWLEDGE STRUCTURES

In this part, knowledge is treated just as the one describing qualitative structural relationships, promoting or suppressing, among problem elements without considerations on problem substances. The methodologies for an integration of the knowledge for quantifying causal relationships is performed graph-theoretically and knowledge retrieval system is presented on a basis of its structural properties. Using these methodologies, individual decisionmaker's conformity for the alternatives with respect to the decisionmaker's psychological stress is measured and a DSS is constructed that recommends the policy alternatives accordingly in an interactive manner.

CHAPTER 3

CONSTRUCTION OF KNOWLEDGE BASE USING COGNITIVE MAPS

3.1 Introduction

Recent highly complex societal problems involve the interrelationships between various aspects of society such as economics, environment, and politics. The ability to ameliorate complex problems requires that the interrelationships be identified and represented. The knowledge used for the identification and representation is mainly extracted from expertises in various fields and is growing more interdisciplinary in nature and becoming greater in amount. Such knowledge should be well-organized so that decisionmakers can access and apply it effectively to real-world problems.

In recent years computer memory hardware and software have made great progress, and it has been possible to store vast amounts of information in computer memory. It is likely that all kinds of information will be stored in it, which includes not only numerical data but also documentary data such as reports, papers, and minute books. Especially in decisionmaking the documentary data written by experts, e.g., diplomats, economists, and administrators are an important source of various expertises. Such documentary data describe the relationships among social aspects. Therefore, if we are to use the documentary data by storing them in a computer, the data should be organized so that the relationships are represented explicitly.

The relationships described in documents on societal problems are mainly causal relationships between various elements of society. In this chapter, we first propose a knowledge representation in computers using *cognitive maps*, which were proposed by R.Axelrod and others in international politics and are an attempt to utilize policy experts' beliefs and cognition about foreign policy for policy analysis, forecasting, decisionmaking, and so on (e.g., Axelrod, 1976; Hart, 1977; Eden and Jones, 1980; Klein and Cooper, 1982). The constituents of cognitive maps are nodes called *concepts* and signed arrows representing *causal relations* between concepts. The policy alternatives, all of the various causes and effects, the goals and the ultimate utility of the decisionmaker can all be adopted as concepts. The arrows are also encoded according to experts' causal assertions. The procedure of derivation of cognitive maps from documents is called *documentary coding*. Then, using it, we constructed a knowledge base system for decision support in which all causal structures described in many documents are joined together. The processes of coding cognitive maps from documents for constructing knowledge base are dicussed with the results of verifying the coding reliability in this chapter. Furthermore, considerations on knowledge base management in its construction and modification processes are also made (cf. Nakamura, Iwai and Sawaragi, 1982a, 1982b).

3.2 A Review of Cognitive Maps

3.2.1 Derivation of Cognitive Maps

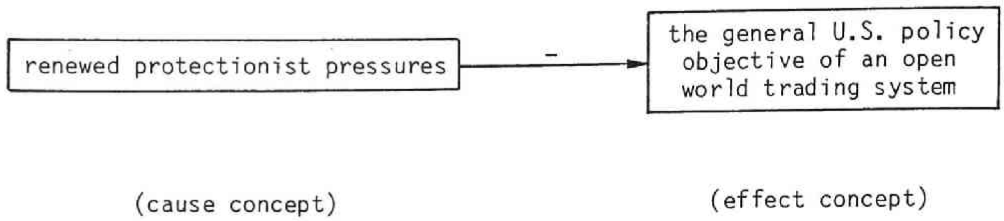
The idea that relationships among concepts can be systematically derived with high fidelity from documents is based on "Evaluative assertion analysis" by C.E. Osgood, S. Saporta, and J.C.Nunnally (Osgood, Saporta and Nunnally, 1956).

Documentary coding is, as a rule, transformation of causal assertions in documents into causal relations consisting of cause concepts, effect concepts, and signed arrow. The concepts must be represented as variables which have the potential to take on different values, for example "seculity of Japan" and "consensus of inhabitants." Entities such as "Japan" and "habitants" are not concepts since they can take on no other values. The value of concepts is not necessarily continuous or metrical but may be a categorical one such as "large" and "little." The sign of an arrow is determined as plus when the causal assertion states that the cause concept increases or promotes the effect concepts and as minus when it states the contrary. The codiong is rather easy when the sentence consists of a definite subject, object, and verb. For example, the following sentence,

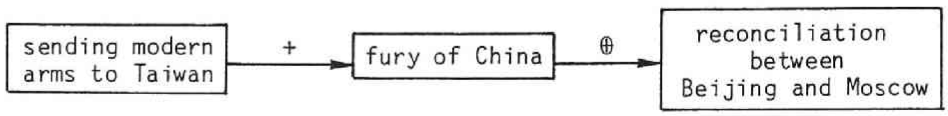
"Renewed protectionalist pressures also threaten the general U.S. policy objective of an open world trading system,"

is coded as shown in Fig.3.1(a). In general sentences, however, causal assertions are not so explicitly stated as above and so we must code causal relation according to context. For example, the following sentences can be coded as shown in Fig.3.1(b) (the sign $\oplus$ in this figure is explained subsequently).

"Why infuriate China by sending modern arms to Taiwan, the allies ask, and risk a reconcillation between Beijing and Moscow?"



(a)



(b)

Fig. 3.1 Coding examples of causal assertions

3.2.2. Mathematics of Cognitive Maps

Although we considered only the two kinds of causal relationships, plus and minus, in the previous section, Axelrod and others regarded the relationship in which the cause concept does not affect the effect concept at all as a kind of causal relation as well. They considered that there are the following eight types of causal relations consisting of the the above three elementary causal relations.

A $\xrightarrow{+}$ B: A has plus effect on B.

A $\xrightarrow{-}$ B: A has minus effect on B.

A $\xrightarrow{0}$ B: A has no effect on B.(Generally, this relation is not depicted in the cognitive maps.)

A $\xrightarrow{\oplus}$ B: A will not hurt or does not prevent B. Namely,

$$= 0 \cup + = [0, +]$$

$A \xrightarrow{\theta} B$: A will not help or does not promote B. Namely,

$$\theta = 0 \cup - = [0,\ -]$$

$A \xrightarrow{m} B$: A has any effect on B. Namely

$$m = + \cup - = [+,\ -]$$

$A \xrightarrow{u} B$: All the three elementary relations can exist between A and B. Namely,

$$u = m \cup 0 = [+,\ -] \cup 0 = [+,\ -,\ 0].$$

This relation is called *universal*.

$A \xrightarrow{a} B$: The two relations plus and minus, which are contrary to each other, are asserted. Namely,

$$a = + \cap - = \emptyset.$$

This relation is called *ambivalent*.

As above, the eight types of causal relations are represented by sets of the three elementary causal relations, plus, minus, and zero. Causal relationships described by several causal assertions are determined based on the set theoretical operations.

(1) Combining assertions to form relationships

When two assertions whose signs are x and y are described between cause Concepts A and effect Concept B as shown in Fig.3.2(a), Axelrod and others proposed intersection of x and y as the combined relationship of the two assertions. This operation means to adopt only the relationship common to both the relations. For example, if $x = \oplus$ and $y = +$ as shown in Fig.3.2(a), the relationship between A and B is determined as follows. $x \cap y = \oplus \cap + = [0,\ +] \cap + = +$. Namely, the relationship is coded as "A helps B" in the cognitive map.

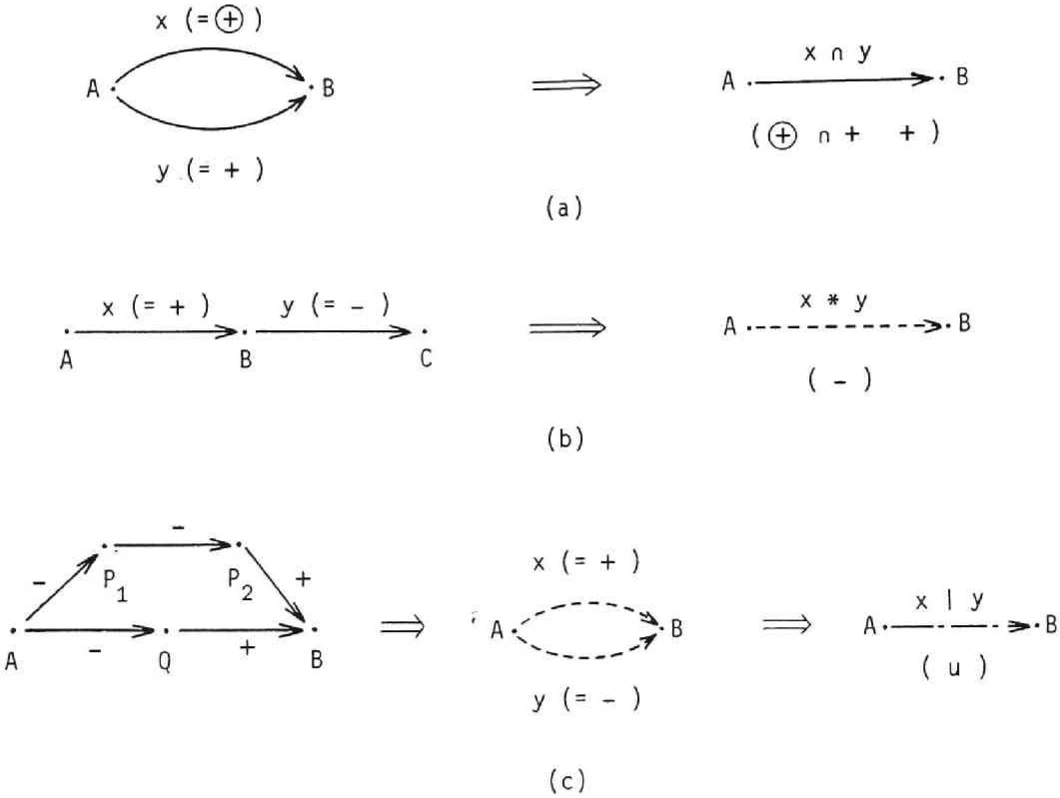


Fig. 3.2 Mathematics of cognitiveb maps.

(2) Calculating the indirect effect of a single path

Suppose there is a direct effect, i.e., causal assertions, x from Concept A to Concept B and a direct effect y from Concept B to Concept C through B. This path carries an indirect effect from A to C. The indirect effect is defined by the following multiplication $x*y$:

$$x*y = y \text{ if } x = + \text{ and } y \in R \quad (3\text{-}1)$$

$$x*y = 0 \text{ if } x = 0 \text{ and } y \in R \quad (3\text{-}2)$$

$$x*y = a \text{ if } x = a \text{ and } y \in R / 0 \quad (3\text{-}3)$$

$$x*y = + \text{ if } x = - \text{ and } y = - \quad (3\text{-}4)$$

where R is the set of all the eight types of causal relations, i.e., R = [+, -, 0, $\oplus$, θ, m, u, a].

multiplication distributes over union (e.g., $(-)*(\theta)$ =

$= (-)*(0 \cup -) = ((-)*(0)) \cup ((-)*(-)) =$

$= 0 \cup + = \oplus$ (3-5)

multiplication is symmetric. (3-6)

These six rules seem to be reasonable. For example, Rule (3-2) says that, if A has no effect on B ($x = 0$), it is natural that A has no effect on C ($x*y = 0$) even if there is any effect from B to C ($y \in R$). Rule (3-3) says that, if the effect from A to B is ambibalent ($x = a$), the indirect effect from A to C cannot help being ambivalent ($x*y = a$) even if the effect from B to C may be determined as any effect ($y \in R / 0$).

(3) Determining the total effect of different paths

When there are two paths from Concept A to Concept B as shown in Fig.3.2(c), the total effect of from A to B is as follows:

$x|y = y$ if $x = 0$ and $y \in R$ (3-7)

$x|y = a$ if $x = a$ and $y \in R$ (3-8)

$x|y = x$ if $x = y \in [+, -]$ (3-9)

$x|y = u$ if $x = +$ and $y = -$ (3-10)

addition distributes over union (e.g., $(+)|(\theta)$

$= (+)|(0 \cup -) = ((+)|(0)) \cup ((+)|(-))$

$= + \cup u = u$) (3-11)

addition is symmetric. (3-12)

These rules are also easily acceptable except Rule (3-8). In regard to Rule (3-10), the total effect depends on which of the indirect effects (x =

+ and $y = -$) is stronger, although it is ambivalent if both x and y are direct effects as defined before. The total effect is plus if indirect effect x (= +) is stronger than indirect effect y, and is zero if x is as strong as y. Therefore $x|y$ is $x \cup - \cup 0 = [+, -, 0] = u$. Rule (3-8) is considered to be based on the idea that if an ambivalent effect is included in the indirect effects, that effect is prior to other effects in determining the total effects.

3.3 Construction of Knowledge Base

3.3.1 Combination of Cognitive Maps

Experts' documents concerned with a complex problem contains expertise from their own and different points of view and the cognitive maps derived from the documents are different in concepts. However, it is natural that several concepts are common to some maps as far as the documents are concerned with the same problem. Using the common concepts, we join the cognitive maps to form a knowledge base as shown in Fig.3.3.[1] The common concepts play a role of relaying nodes in joining. We coded cognitive maps of five documents relevant to "traffic problems in Japan" and constructed a knowledge base concerning the problem. The knowledge base consists of 152 concepts and 265 direct causal relations.

The knowledge base is considered an integration of various cognitions concerning the problem and can provide the decisionmakers with such overall and detailed information about the problem as follows.

---

1 We considered the two elementary causal relations, positive and negative, in coding of causal assertions. In this and subsequent sections, arrows without sign denote positive causal relations and only arrows with minus sign denote negative causal relations.

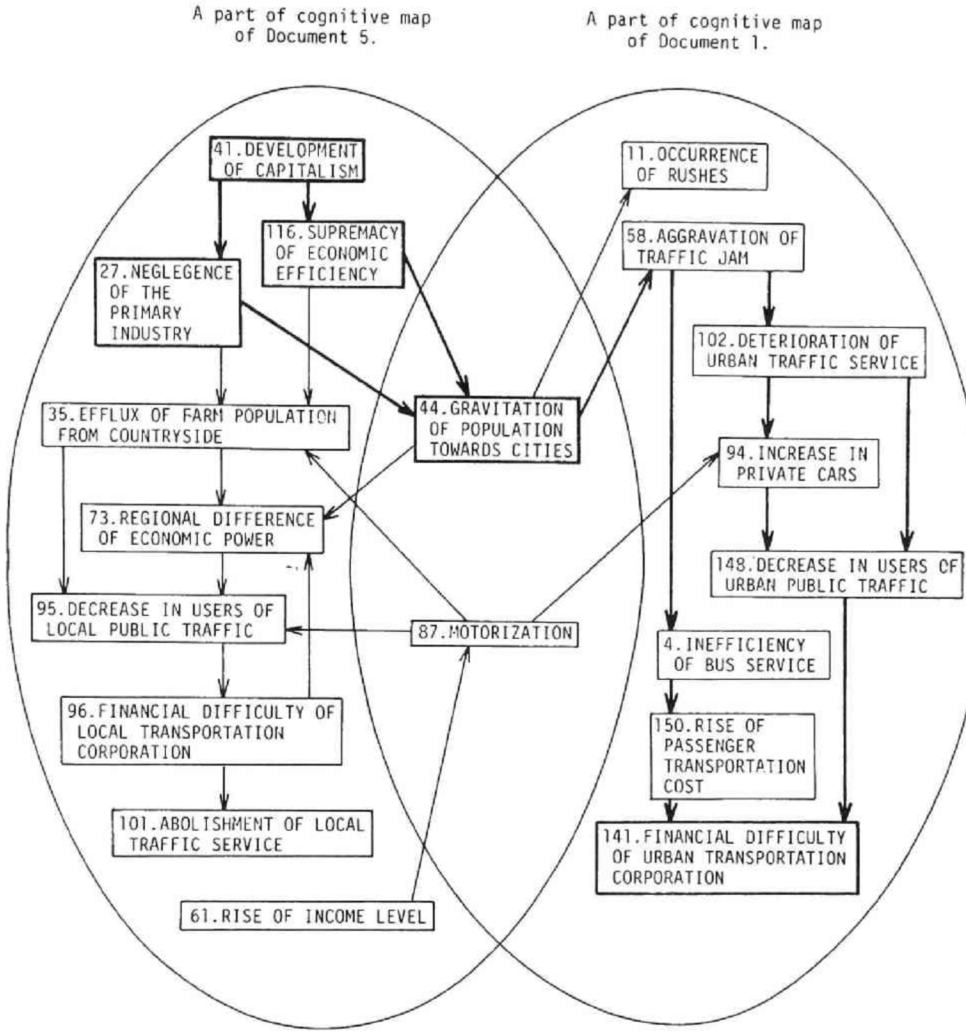


Fig. 3.3 Joining of cognitive maps to form knowledge base

(1) Expansion of causal structure by supplementation of intermediary concepts

Between Concepts "A4—motorization" and "A17—increase in automobile traffic," only a direct causal relation is asserted in Document 1, as shown in Fig.3.4(a). However, by adding the cognitive map of another Document 3,

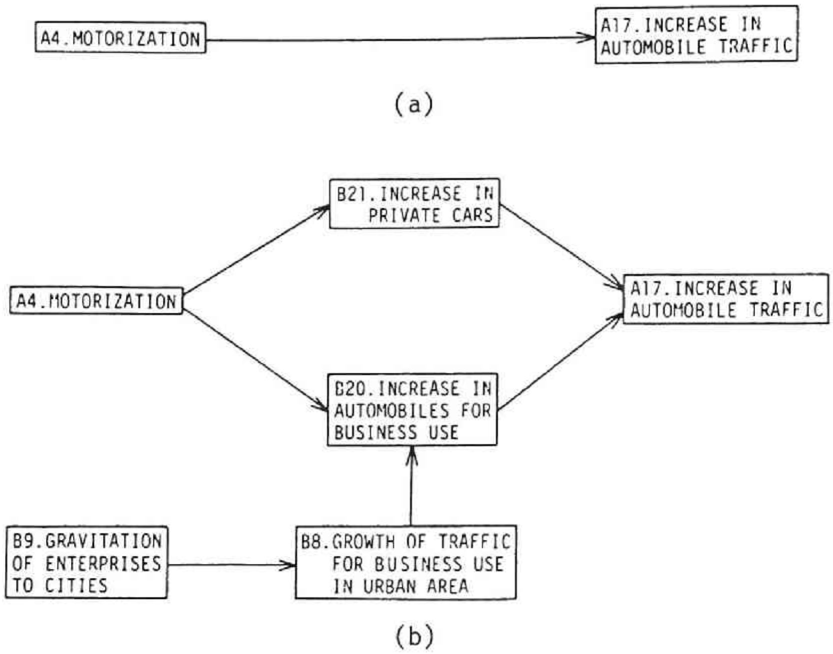


Fig. 3.4 Examination of causal structure by supplementation of intermediary concepts in the process of joining cognitive maps.

two intermediary Concepts B20 and B21 are inserted into the above causal relation as shown in Fig.3.4(b). By the insertion of concepts, we can recognize that "A4—motorization" promotes increase in two kinds of automobiles, private and business use, and that "increase in automobiles for business use (B20)" is also promoted by "gravitation of enterprises to cities (B9)" through "growth of traffic for business use in urban area (B8)." As above, supplementation of intermediary concepts expatiates causal structure of cognitive maps and makes causal relations asserted by an expert easier for decisionmakers to grasp.

(2) Derivation of long causal path from one cognitive map to another

Fig.3.3 shows parts of cognitive maps derived from Documents 1 and 5. While Document 1 is only concerned with *urban traffic*, Document 5 describes *traffic related to local problems*. As shown in the map of Document 5, the author of the document indicates that "44—gravitation of population towards cities" is caused by "41—development of capitalism" through "116—supremacy of economic efficiency" and "27—negligence of the primary industry." In the map of Document 1, its author asserts that "44—gravitation of population towards cities" forms a cause of "141—financial difficulty of urban transportation corporation." The two indirect causal relations include Concept 44 in common. Consequently, Concept 44 plays the role of relaying node connecting the two indirect relations in joining the two cognitive maps and a new long causal path is formed from Concept 41 to Concept 141. The long causal path gives us information that "41—development of capitalism" is a remote cause of "141—financial difficulty of urban transportation corporation." Apparently, the information cannot be derived from the map of Document 1 nor 5 but only from the joined map. Thus, in the joined map, cognition of various experts (authors) are integrated and an overall grasp of the *traffic* problem is constructed, which each of the experts can never hold.

The above two kinds of useful information in this knowledge base of joined cognitive maps seem to be usually generated by decisionmakers in their minds when they read experts' documents and integrate experts' knowledge. The knowledge base can help decisionmakers to gain such an integrated grasp of the problem without reading experts' documents.

3.3.2 Management of Causation Knowledge Base

Terminologies used in documents vary with the authors. The same notion may be represented by various terms. Therfore when we join a cognitive map of a new document to the knowledge base, we find difficulties in such a construction of large knowledge bases how to find a concept represernting the same notion as a concept of the new map from the knowledge base. We propose the following two methods to make it easier to retrieve the corresponding concepts in content from the knowledge base.

(1) Search method with key words

A coder selects a key word representing a new concept and enters it to the system. A concept is represented by a string of several words. The system prints out all the concepts including the key words from the knowledge base. Suppose a new concept is added, such as " drift of population towards cities," this concept is not word-for-word identical with Concept "44—gravitation of population towards cities" in the knowledge base but is nearly the same as it in content. The coder selects population as a key word representing the new concept and enters it. The system prints out the following three concepts: "44—gravitation of population towards cities," "120—sprawling of population in cities" and "64—security of public service in depopulated area." Consequently, the coder easily finds Concept 44 corresponding to the new concept. This system can deal with some inflections and synonyms. For example, in the above case, Concept 64 is printed out because it contains the term "depopulated" instead of the key word "population." Also, since "car" and "wheels" are synonyms for "automobile," concepts including the formal two key words are all printed out by search with the key word "automobile."

(2) Search method by category of concepts

Some concepts agree with each other in content though no term is common to them. For example, a concept "influx of people into urban area" has no term common to Concept 44 but represents a closely similar notion to it. To deal with such a correspondence between concepts, we propose a search method by categories of concepts. We set several categories and

1. PUBLIC TRANSPORTATION	1. 2. 3. 4. 5. 6. 7. 8. 9. 10. 11. 14. 22. 28. 46. 51. 53. 54. 55. 56. 60. 67. 86. 95. 96. 100. 101. 102. 105. 107. 108. 110. 117. 121. 125. 127. 136. 141. 142. 148. 149. 150.
2. AUTOMOBILE TRANSPORTATION	4. 5. 12. 13. 29. 34. 58. 59. 60. 67. 72. 78. 79. 80. 85. 87. 91. 92. 94. 97. 98. 99. 103. 139. 140. 151. 152.
3. FACILITIES FOR TRANSPORTATION	16. 20. 21. 31. 47. 63. 74. 106. 111. 112. 113. 123.128.132.133.134.146.
4. QUALITY OF LIFE	2. 10. 11. 44. 47. 57. 61. 84. 85. 90. 94. 101. 102. 103. 104. 105. 107. 114. 115. 118. 119. 122. 124. 130. 144. 145. 146. 152.
5. ENVIRONMENT	19. 81. 82. 83. 89. 103. 109. 122. 124.
6. ECONOMY	18. 23. 24. 25. 27. 33. 38. 39. 41. 61. 68. 71. 73. 115. 116. 123.
7. LOCAL PROBLEM	21. 24. 26. 32. 35. 37. 46. 50. 64. 73. 76. 88. 95. 96.101.123.128.149.
8. SAFETY	67. 80. 113. 126.
9. URBAN PROBLEM	15. 44. 52. 63. 68. 75. 76. 103. 119. 120. 141. 144.147.148.
10. INDUSTRY	26. 27. 30. 38. 39. 42. 43. 45. 48. 49. 62. 65. 66. 69. 70. 71. 72. 75. 76. 82. 85. 93. 97. 98. 129. 135. 137. 143.
11. TECHNOLOGY	48. 49. 69. 138.
12. POICY	22. 50. 72. 92. 99. 109. 117. 127. 129. 135. 140. 151.
13. CULTURE	37. 57. 83.
14. DISTRIBUTION	77. 78. 79. 80. 110. 118. 151. 152.
15. THE OTHERS	17. 36. 40. 131.
16. THE WHOLE	1.– 152.

Table 3.1 Classification of concepts in the implemented knowledge base

classified all the concepts included in the knowledge base into the categories. Each category is relevant to the aspect of the problem (traffic problem in Japan). Table 3.1 is a list of classified concept numbers. In our knowledge base, the concepts are classified into fifteen categories. A concept may be included in several categories. For example, Concept 44 is put into categories: urban problem and quality of life. When the coder searches a concept corresponding to the above concept "influx of people into cities," he selects a category closely related to the concept from the fifteen categories, for example "urban problem" and looks over the concepts of the category to find the corresponding Concept 44.

After it is found that there is a concept corresponding to a new concept, the concepts are merged and all the direct relations incident ain and out of the new concept are added to the corresponding concept in the knowledge base. When corresponding concepts are not found, new concepts are added to the knowledge base and entered in some categories. The two methods may fail to find concepts corresponding to some new concepts though the corresponding concepts have already been stored in the knowledge base. Consequently the knowledge base includes the two concepts representing a very similar notion. In such cases the long causal paths which corresponding concepts should relay are left unconnected, and so the knowledge supplemented by the connection is missed.

We constructed the knowledgge base using a network model of database because the cognitive map is a diagraph. The computer is FACOM-M385 and the program language is PL/I. The database stores a network representing reachability between the concepts as well as the joined cognitive map since the data of reachability is frequently used in the retrieval of relationship mentioned in the next section.[2]

---

[2] Modification of the joined cognitive map does not require overall changes in the network of reachability but only partial modification of the network.

3.3.3 Considerations on Reliability of Cognitive Maps

The process of documentary coding is done based on coders' judgment. Therefore, according to differences between coders' subjective judgments, it is feared that there can be some difference between the cognitive maps of the same document derived by the different coders. It is desirable that such difference should be minimized if we are to use documentary coding as a reliable method of deriving the cognitive maps.

In this section, to test the reliability of the coding method (that is, the capacity of the method to yield the same results), we compare two cognitive maps of the same document coded by two different coders and investigate the effect of coders' subjective judgment on the derived cognitive maps. Let x and y denote the two coders, and X and Y denote the maps coded by x and y, respectively.

(1) Agreement in extraction of concepts

We first examine the differences in choice of concepts. Some concepts agree word-for-word with those of the other cognitive map. However, such concepts are not many. Other concepts agree in content with those in the other map, though some expressive difference is involved in the agreement, as in Example 1.

Example 1:

gravitation of pivotal administrations to cities	→	high condensation of pivotal administarions in cities

Some others agree if the sign of causal relations linked to them is changed to the contrary, as shown in Example 2.

Example 2:

aggravation of a traffic jam $\xrightarrow{-}$ easing a traffic jam

These differences are because the two coders derived the concepts representing the same notion from different causal assertions in the documents.

The *concept-to-concept (CC)* agreement consists of the above three cases. Besides CC agreement, there can be *concept-to-set of concepts (CS)* agreement and *set of concepts-to-set of concepts (SS)* agreement.

Fig.3.5 shows parts of cognitive maps X and Y and indicates two examples of CS agreement. While coder x chose one concept "X17—extension of a zone of life," coder y chose "Y24—extension of an urban district" as well. This is because coder y considered that there was some difference between the above two concepts in content, while coder x thought that they represent the same notion. Also, coder x regarded the notion which coder y separated into two concepts "Y26—increase in distance of communication" and "Y34—increase in long distance commuters" as a single concept "X14—increase in long distance communication."

If we consider the sets of concepts in the map Y corresponding to the concepts in map X, the structures of direct relations of maps X and Y agree. However, path 120 -> X17 -> X14 -> 28 in map X does not satisfy the transitivity of indirect relation as shown in map Y. That is, map Y includes direct relations from set of Concepts 145 to set of Concepts 146 and from 146 to Concept 28 which correspond to direct relations from Concept X17 to Concept X14 and from X14 to Concept 28, respectively. However, the direct relations from 145 to 146 reach only Concept Y26 in 146 and the direct relation from 146 to 28 starts only from Concept Y34 in 146. Consequently, coder y did not recognize the transitivity of path 145 -> 146

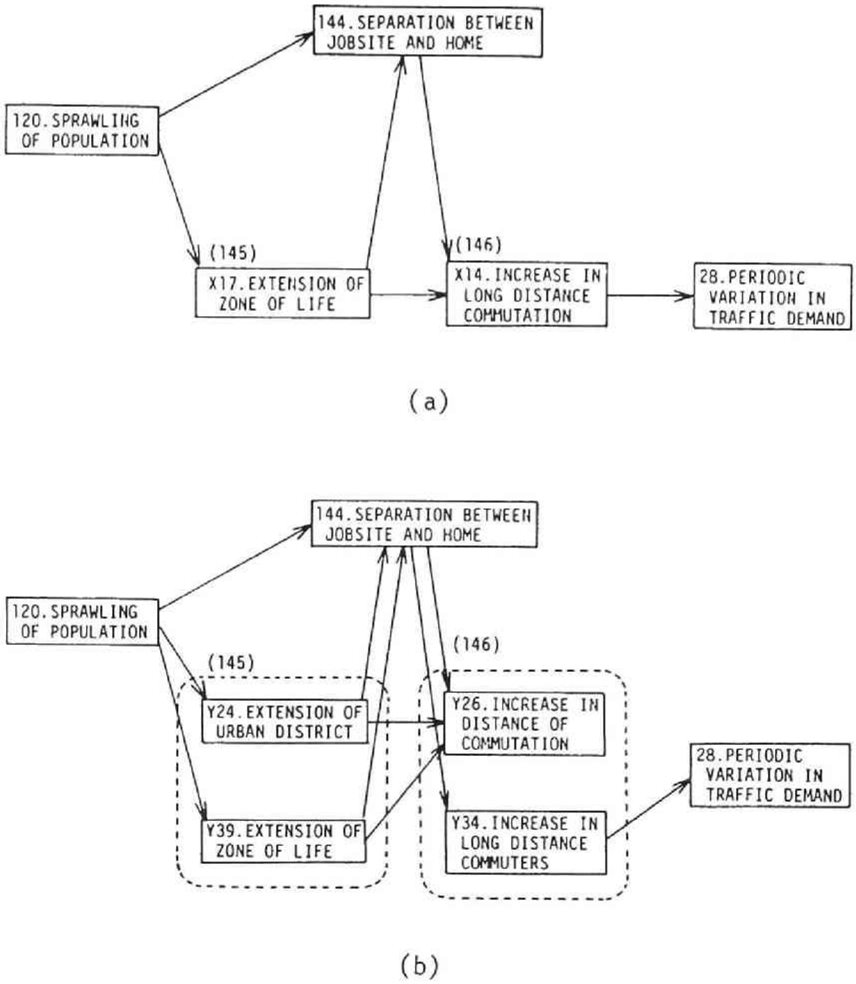


Fig. 3.5 Expatiation of causal structure by supplementation of intermediary concepts in the process of joining cognitive maps

-> 28 which coder x depicted since he put Concepts Y26 and Y34 together into a single Concept X14. Thus, cognitive maps may include some indirect relations which do not satisfy the transitivity though they are very rare if authors' assertions in documents are consistent.

We define the following index as degree of agreement in choice of concepts:

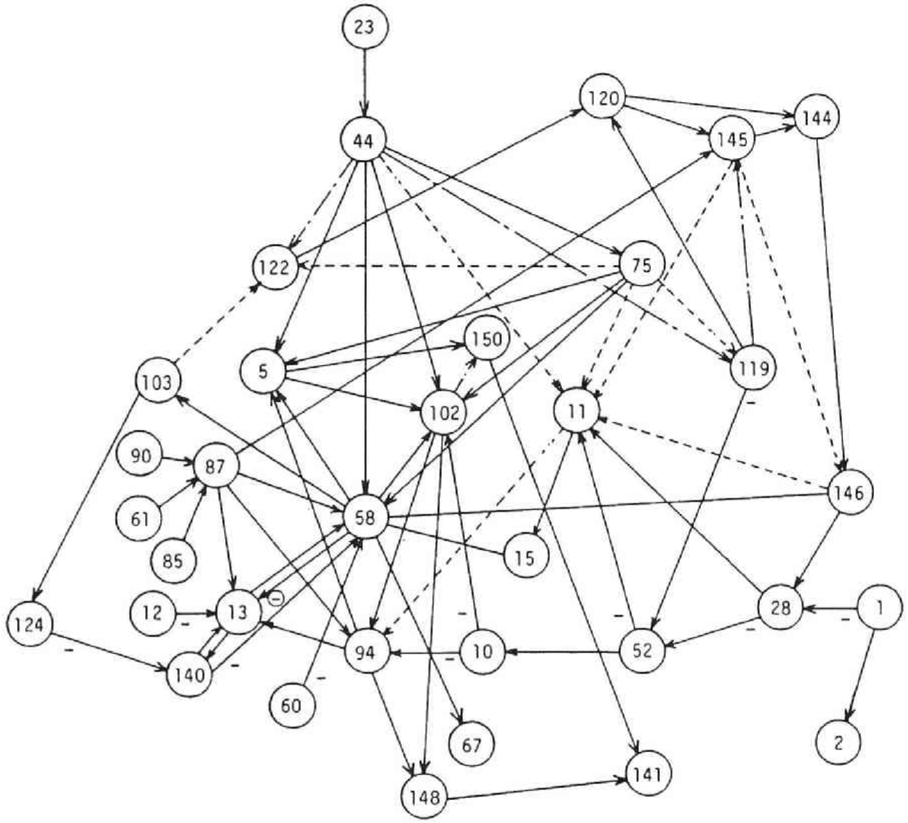
$$s = \frac{X_Y^{\#} + Y_X^{\#}}{X^{\#} + Y^{\#}} \tag{3-13}$$

where X and Y are the sets of concepts in maps X and Y, respectively. X_Y and Y_X denote the sets of concepts of maps X and Y which agree with those in the other maps Y and X, respectively. $X^{\#}$ denotes the number of concepts in map X. $X^{\#}$ and $Y^{\#}$ are 45 and 46, respectively. CC agreement includes 29 pairs; that is, $X_Y^{\#}$ (= $Y_X^{\#}$) is 29. Consequently, the index of CC agreement is $s = (2 \times 29)/(45 + 46) = 0.64$. The number of concepts of maps X and Y included in CS agreement are ten and ten, respectively; that is, both $X_Y^{\#}$ and $Y_X^{\#}$ are ten in this coding example. The index of CS agreement is $s = (10 + 10)/(45 + 46) = 0.22$. This coding example includes no SS agreement. The total agreement is 0.86 (= 0.64 + 0.22). The rest, 0.14, consists of intermediary concepts chosen by one coder and concepts contained in ambiguous causal assertions.

(2) Agreement in extraction of causal relations

To compare causal relations of the two cognitive maps X and Y, the disagreed concepts, i.e., the intermediary concepts and the concepts derived from the ambiguous causal assertions are excluded from the maps. The causal relations relayed by the intermediary concepts are held. Each set of concepts in CS agreement is represented by its corresponding concepts. The maps composed as above are identical with each other in concepts. Fig.3.6 shows agreement in causal relations derived by both the coders and the other kinds of lines denote those derived by one coder x or y.

We propose the following index as degree of agreement in direct causal



1. PEAK ROAD PRICING
2. CHARGE TO USERS
5. INEFFICIENCY OF TRAFFIC SERVICE
10. INCREASE IN FACILITIES FOR PUBLIC TRANSPORTATION
11. OCCURRENCE OF RUSHES
12. SELF-IMPOSED CONTROL OF CAR USAGE
13. INCREASE IN AUTOMOBILE TRAFFIC
15. PARALYSIS OF URBAN FUNCTIONS
23. HIGH GROWTH OF JAPANESE ECONOMY
28. PERIODIC VARIATION IN TRAFFIC DEMAND
44. GRAVITATION OF POPULATION TOWARDS CITIES
52. EXPANSION OF PUBLIC TRANSPORTATION CAPACITY
58. AGGRAVATION OF TRAFFIC JAM
60. TRAFFIC CONTROL SYSTEM OVER A LARGE AREA
61. RISE OF INCOME LEVEL
67. FREQUENCY OF TRAFFIC ACCIDENTS
75. GRAVITATION OF PIVOTAL ADMINISTRATION TO CITIES
85. STABILIZATION OF AUTOMOBILE PRICES
87. MOTORIZATION
90. CONSCIOUSNESS OF BELONGING TO MIDDLE CLASS
94. INCREASE IN PRIVATE CARS
102. DETERIORATION OF URBAN PUBLIC TRAFFIC SERVICE
103. AGGRAVATION OF TRAFFIC POLLUTION
119. RISE OF LAND PRICE
120. SPRAWLING OF POPULATION IN CITIES
122. AGGRAVATION OF LIVING ENVIRONMENT
124. CONSCIOUSNESS OF INHABITANTS
140. IMPROVEMENT OF ROADS
141. FINANCIAL DIFFICULTY OF URBAN TRANSPORTATION CORPORATIONS
144. SEPARATION BETWEEN JOBSITE AND HOME
145. EXPANSION OF ZONE OF LIFE
146. INCREASE IN LONG DISTANCE COMMUTATION
148. DECREASE IN USERS OF URBAN PUBLIC TRAFFIC
150. RISE OF PASSENGER TRANSPORTATION COST

———— : direct causal relation explicitly derived by both the coder x and y.
—·—·— direct relation implicitly recognized by both the coders.
- - - - - : direct relation recognized by one coder x or y.

Fig. 3.6 Agreement of in extraction of causal relations from document by different two coders x and y.

relations in the same way as (3-13):

$$t = \frac{2 \times (X \cap Y)^{\#}}{X^{\#} + Y^{\#}} \tag{3-14}$$

where X and Y are the sets of causal relations derived by coders x and y, respectively. $X^{\#}, Y^{\#}$, and $(X \cap Y)^{\#}$ of Fig.3.6 are 58, 63, and 54, respectively. Consequently $t = (2 \times 54)/(58 + 63) = 0.89$.

Some of the direct relations derived by one coder are implicitly recognized by the other coder. For example, the direct relation 44 -> 122 by coder x is implicitly recognized by coder y as indirect relation 44 -> 75 -> 122. Therefore, the causal relation from 44 to 122 is concluded whichever map derived by coder x or y may be adopted. Those relations amount to eight: 44 -> 122, 44 -> 119, 44 -> 11, 145 -> 11, 145 -> 146, 146 -> 11, 119 -> 145, and 102 -> 145. If those relations are considered to be recognized by both coders, degree of agreement in causal structure is fairly high. The five relations 75 -> 122, 75 -> 119, 75 -> 11, 11 -> 94, 103 -> 122, which do not agree consist of the following: those derived from ambiguous causal assertions in the documents and those coded dogmatically and intuitively by coders without basis in the context of the document.

As a result, if coders do not adopt ambiguous assertions in documents and do faithful coding according to the context of documents, the agreement in extraction of concepts and causal relations will be improved and the documentary coding can be used as a reliable method of derivation of the cognitive maps.

3.4 Concluding Remarks

In this chapter, we first summarized the method of documentary coding and derivation of cognitive maps. Second, we constructed a knowledge base by joining cognitive maps of five documents relevant to the traffic problem in Japan and indicated that some kinds of useful information are generated in the knowledge base through the process of joining. Then, in order to let the construction process of the knowledge base be smooth and easy, the management methods for the knowledge base were proposed for retrieving similar concepts from the knowledge base. Finally, we investigated two coders' results by the method and confirmed the sufficient reliability of the documentary coding method for derivation of cognitive maps.

As mentioned in Section 3.3.3, the problems related to the duplication of similar concepts and the transitivity of causation must be solved for implementation of advanced decision support systems with deductive inference mechanism. Though the documentary coding is coder's manual labor, it will be necessary to develop automated coding systems based on techniques for natural language processing in order to deal with a large number of documents.

The knowledge base presented here is a particular type of semantic network dealing only with causation between concepts. However, it is necessary to organize other kinds of relations described in documents (e.g., sentiment relation and unit formation (Heider,1946; Katai & Iwai, 1978)) into a semantic network, so that in the decision support of real-world problems the knowledge base will serve as an intelligent interface between the decisionmakers and the documentary data stored in computer memory.

CHAPTER 4

DECISION SUPPORT FOR SUCCESSIVE FOCUSING PROCESS AND QUANTIFICATION OF CAUSAL EFFECTS BASED ON CAUSATION KNOWLEDGE BASE

4.1 Introduction

With respect to a knowledge based decision support system for ill-structured problems, knowledge is to be clarified and presented to a decisionmaker in advance of its utilization in the further inferences. It is because a decisionmaker facing such a complex problem cannot help feeling much procedural uncertainty, that is, about what knowledge, namely, what information from experts, he should seek for, and about how to get the relevant knowledge from an immense amount of it in both efficient and comprehensive ways.

Then, what is required of DSS is to help the decisionmaker focus his subject of interest in the vast amount of available knowledge, and to gradually or successively deepen his understanding of the problem, from global aspects to specific ones and from simplified views to detailed ones, and so on. Since the causal knowledge in the knowledge base, when they are elicited from different sources, is so often conflicting, it makes the scope of uncertainties much greater. Therefore, for DSS it is essential to enable him to understand those conflicts of opinions just as they are, rather than to resolve such conflicts in some black-boxed way.

In this chapter, for the knowledge base constructed in the previous chapter, we will develop knowledge retrieval facilities enabling the decisionmakers to interactively retrieve relevant information from the knowledge base and form their overall and detailed understanding of the complex problems based on the expertise stored in the knowledge base according to their individual viewpoints in the decision process (cf. Nakamura, Iwai and Sawaragi, 1982a, 1982b). Next, we develop measures which enable quantitative evaluations of uncertainties contained in the practical consequences of each alternative by evaluating to what degree it is favourable and unfavourable to the user's goals based on causal relationships stored in the knowledge base (cf. Sawaragi, Iwai and Katai, 1984, 1985b, 1986a, 1986b).

4.2 Support for Successive Focusing Processes Using Causation Knowledge Base

4.2.1 Three Retrieval Modes Corresponding to Types of User's Requests

The knowledge base contains many kinds of concepts and causal relations between them. For real decision support using the knowledge base, we propose a retrieval system to present relevant information from the knowledge base to decisionmakers according to their requests.

The retrieval system has the following three retrieval modes according to types of user's requests.

(1) Presentation of skeleton maps

The system displays skeleton maps comprising some significant concepts and causal relations between them, relevant to the user's subject of interest. Axelrod and others regarded *centrality* of concepts in cognitive maps as the significance of concepts (Axelrod, 1976). This centrality is measured by *total degree* of concepts. The total degree of a concept is the number of direct relations incident in and out of the concept. In the implemented system, not more than eight of the most significant concepts (according to the centrality and causal relations between them) are displayed. The sign of the relations in the skeleton map is that of the shortest paths between the concepts.[1] The user selects one of the three codes: A(category), B(key words), and C(documents), and enters it to the system.

A(category): the user chooses a category relevant to his subject of interest from the before-mentioned 15 categories and a category "whole traffic problem," and the system selects the significant concepts from the category (cf. Table 3.1).

B(key words): the user chooses some key words representing his subject of interest and enters them. The system selects the significant concepts containing at least one of the key words.

C(document): the user chooses a document from the list of documents whose cognitive maps are stored in the knowledge base and then the significant concepts are selected from the concepts contained in the cognitive map of the document.

The skeleton maps give the user overall information relevant to his subject of interest and help him to grasp the subject in its totality.

(2) Display of hierarchical graphs of detailed causal structure

When the user wants to know detailed causal structures of causal relations presented in the skeleton map, the system displays the detailed

---

[1] In real problems, indirect relations between concepts are usually universal because there are so many causal paths which are different in sign.

structure, which is printed out in the form of a hierarchical graph, and prints out the graph.

(3) Retrieval of sources of causal assertions

The user sometimes wants to know by whom and how the causal relations in the displayed graph are asserted in source documents, for example when the causal relations are ambivalent and difficult for him to understand or accept. In such cases the system presents the sources of causal relations by document number, page, and line and then the user can browse relevant parts of the sources.

4.2.2 An Example of Interactive Retrieval with the Knowledge Base for Decision Support

Example 1:

Suppose that a decisionmaker wants to know the causal structure of problems concerning "public transportation." The decisionmaker selects code A and enters category 1 (public transportation) to the system to get a skeleton map on public transportation. Fig.4.1 shows the skeleton map printed out by the system. In the figure, the positive arrow from Concept 55 to Concept 10 suggests to the decisionmaker that "demand for transportation of passengers" promotes "increase in facilities for public transportations." When the decisionmaker wants to know why and how the demand increases the facilities for public transportation, he enters Concept 55 and 10 to the system to investigate the detailed causal structure between them. Fig.4.2 shows the hierarchical graph representing the detailed structure. The figure indicates not only a path carrying positive indirect effect 55 -> 9 -> 10 but also two paths carrying negative indirect effect 55 -> 28 -> 3 -> 10 and 55 -> 28 -> 14 -> 10; that is, the

```
WHICH CODE DO YOU CHOOSE? PLEASE SELECT A, B, OR C.
     A.CATEGORY
     B.KEY WORDS
     C.DOCUMENTS
:
A

WHICH FIELDS ARE YOU INTERESTED IN? PLEASE SELECT THE NUMBER.
             1.PUBLIC TRANSPORTATION
             2.ECONOMY
             3.SOCIETY
             4.LIFE
             5.ENVIRONMENT
             6.TRANSPORTATION
             7.AREA PROBLEM
             8.AUTOMOBILE TRANSPORTATION
             9.CITY PROBLEM
            10.INDUSTRY
            11.TECHNOLOGY
            12.POLICY
            13.CULTURE
            14.DISTRIBUTION
            15.SAFETY
            16.NO FIELD
:
1
```

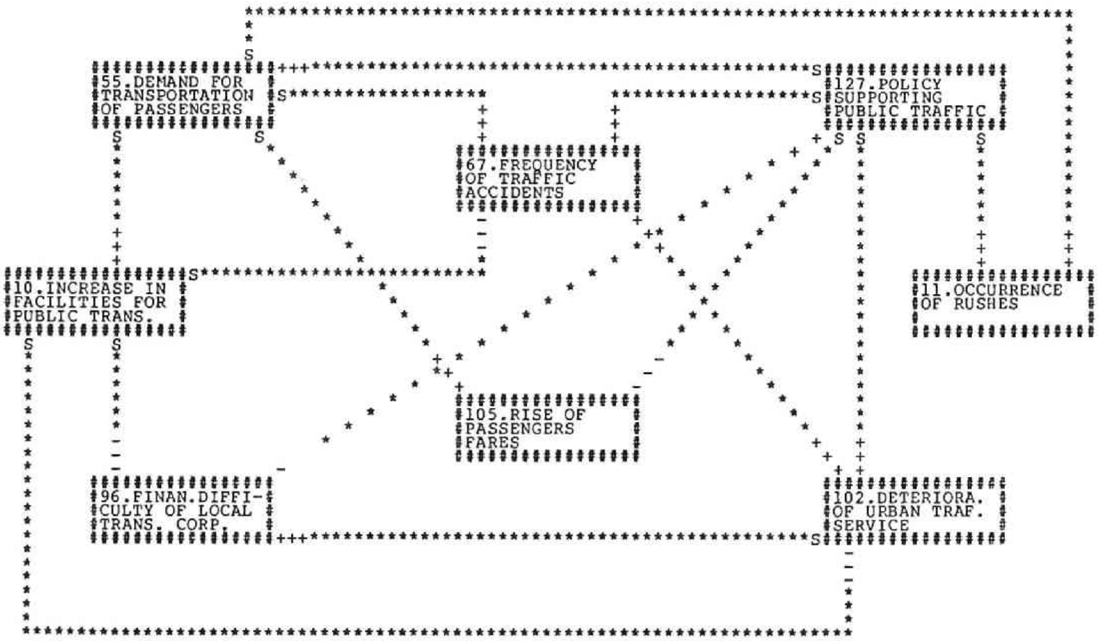


Fig. 4.1 Skeleton map relevant to category PUBLIC TRANSPORTATION: "A s***+++ B" denotes $A \overset{\pm}{\rightarrow} B$, "A ---***+++ B" denotes $A \overset{-}{\rightarrow} B$ and $B \overset{\pm}{\rightarrow} A$.

```
DO YOU WANT TO KNOW DETAILED CAUSAL STRUCTURE OF ANY RELATION?
:
YES
INPUT STARTING NODE
:
55
INPUT ENDING NODE
:
10
```

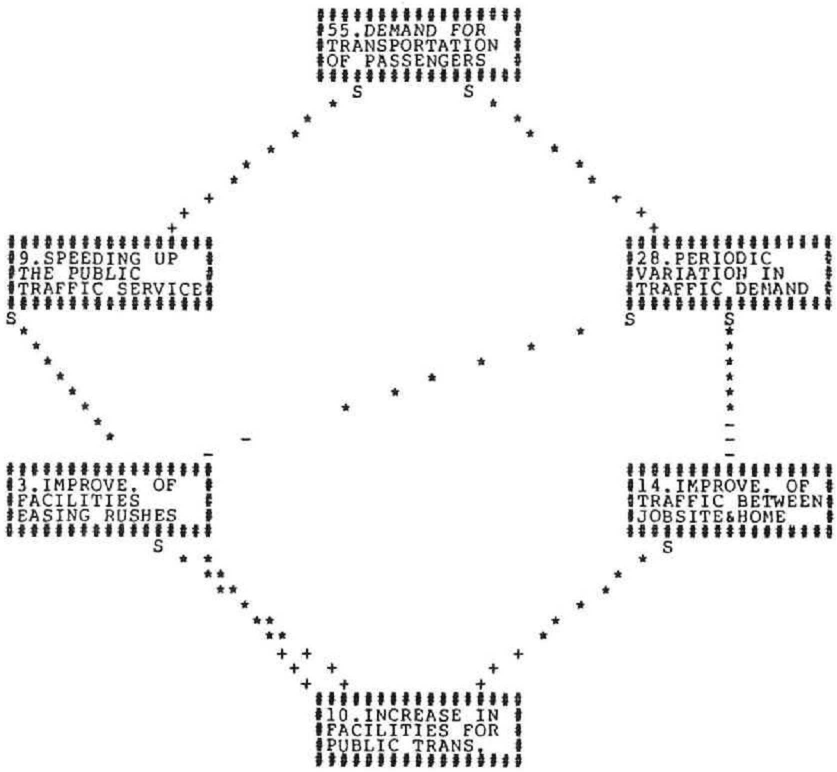


Fig. 4.2 Hierarchical graph of causal structure from Concept 55 to 10 in Fig. 4.1

indirect relation from Concept 55 to Concept 10 is universal. From the figure the decisionmaker can recognize that increase in the demand for transportation promotes "speeding up the public traffic service (9)" and then increases the facilities for public transportation, while the increase in the demand causes "periodic variation of public demand (28)." Then the variation impedes "improvement of facilities easing rushes (3)" and "improvement of traffic between job site and home (14)" and consequently decreases the facilities for public transportation. By entering other significant concepts to the system again, the decisionmaker gets information about them.

As above, the decisionmaker can form his overall and detailed understanding of public transportation through the process of interactive retrieval with the knowledge base. Since the knowledge base is constructed based on causal assertions described in experts' documents, the information indicated in the skeleton maps and hierarchical graphs is also according to the experts' assertions. So far the decisionmaker got it by reading documents for himself. However, those documents are usually so numerous that it is hard for him to read all or even most of them. Our knowledge base system provides the decisionmaker with the information without their reading the documents, although the information is only concerned with causation, and helps them to grasp complex problems based on many experts' knowledge described in their documents.

Also, if the decisionmaker wonders how the universal causal relation in Fig.4.2 is coded, he requests the sources of the direct causal relations composing the universal relation to know by whom and in what context the causal relations are asserted. Fig.4.3 shows the output of sources and indicates that the two direct relations composing the positive path 55 -> 9 -> 10 are asserted in Document 2 while the five direct relations composing the negative path 55 -> 28 -> 3 -> 10 and 55 -> 28 -> 14 -> 10 are asserted in Document 1. According to the printed out document numbers, pages, and lines, the decisionmaker browses the indicated parts of the documents and knows that the author of Document 2 grasps the effect of Concept 55 on Concept 10 only in transportation capacity and considers the effect

```
DO YOU WANT TO KNOW SOURCES OF ANY CAUSAL RELATION?
:
YES
INPUT STARTING NODE
:
 55
INPUT ENDING NODE
:
 28
 55 ***********>  28

     DOCUMENT    1
     PAGE        202
     FROM LINE   1 TO LINE  2

INPUT STARTING NODE
:
 28
INPUT ENDING NODE
:
 14
 28 ***********>  14

     DOCUMENT    1
     PAGE        202
     FROM LINE   9 TO LINE 11

INPUT STARTING NODE
:
  3
INPUT ENDING NODE
:
 10
  3 ***********>  10

     DOCUMENT    1
     PAGE        199
     FROM LINE   6 TO LINE  9

INPUT STARTING NODE
:
  9
INPUT ENDING NODE
:
 10
  9 ***********>  10

     DOCUMENT    1               DOCUMENT    2
     PAGE        206             PAGE        149
     FROM LINE   1 TO LINE  3    FROM LINE   16 TO LINE 19
```

```
INPUT STARTING NODE
:
 55
INPUT ENDING NODE
:
  9
 55 ***********>   9

     DOCUMENT    2
     PAGE        149
     FROM LINE  16 TO LINE 19

INPUT STARTING NODE
:
 28
INPUT ENDING NODE
:
  3
 28 ***********>   3

     DOCUMENT    1
     PAGE        202
     FROM LINE  11 TO LINE 14

INPUT STARTING NODE
:
 14
INPUT ENDING NODE
:
 10
 14 ***********>  10

     DOCUMENT    1
     PAGE        205
     FROM LINE   3 TO LINE  5
```

Fig. 4.3 Output of sources of direct causal relations displayed in Fig. 4.2.

positive while the author of Document 1 considers the effect negative by taking convenience of public transportation into account. The browsing according to information about sources suggests to the decisionmaker that the universal causal relation stems from the difference between the two coders' cognition. As above, our system can inform him which of many documents include the very information he need and which parts he should read in the documents. This means that the third retrieval mode can be used also for an advanced document retrieval method which can find documents concerned with the very subject the user requests.

Example 2:

Furthermore, suppose that the decisionmaker wants to analyze the relationship between two important factors in the traffic problem, service and fare. In such a case, he selects retrieval code B and enters the two key words service and fare to the system. The system prints out a skeleton map whose eight concepts include key words service or fare, as shown in Fig.4.4. In the skeleton map, he focuses on a serious problem of local traffic, i.e., "security of public service in depopulated area (46)" and investigates how the "(national) policy supporting public traffic," which is a significant Concept 127 in Fig.4.1, affects Concept 46 in Fig.4.4. He enters the two concepts and the system prints out a hierarchical graph from Concept 127 to Concept 46 shown in Fig.4.5(a). The graph is found to consist of two causal structures (i) and (ii) shown in Fig.4.5(b). The former structure (i) indicates positive effects of the "policy supporting public traffic (127)" on "security of public service in depopulated area (46)" by easing "financial difficulties of local transportation corporation (96)" and controlling "free competition among traffic corporations (111)." The latter structure (ii) is a causal network mainly carrying negative effects of the policy and includes two pivotal concepts "advancement of industrial structure (42)" and "gravitation of population towards cities (44)." The structure shows that the policy promotes "expansion of transportation capacity (136)" and "modernization (138)," and then causes "the advancement of industrial structure (42)" and "the gravitation of

```
WHICH CODE DO YOU CHOOSE? PLEASE SELECT A, B, OR C.
     A.CATEGORY
     B.KEY WORDS
     C.DOCUMENTS
:
B

INPUT KEYWORDS
:
'FARE'

INPUT KEYWORDS
:
'SERVICE'

105.RISE OF      PASSENGERS       FARES
4.EFFICIENCY     OF BUS           SERVICE
9.SPEEDING UP    THE PUBLIC       TRAFFIC SERVICE
16.REINFORCE.    OF TRAFFIC       SERVICE
46.SECURITY OF   PUBLIC SERVICE   IN DEPOP. AREA
47.TRAFFIC       SERVICE FITTINGLIFE
101.ABOLISHMENTOF LOCAL TRAF. SERVICE
102.DETERIORA.   OF URBAN TRAF.   SERVICE
149.DECREASE INFREQ. OF LOCAL RAILWAY SERVICE
63.UNBALANCE INDEMAND & SUPPLYFOR TRAF.SERV.
```

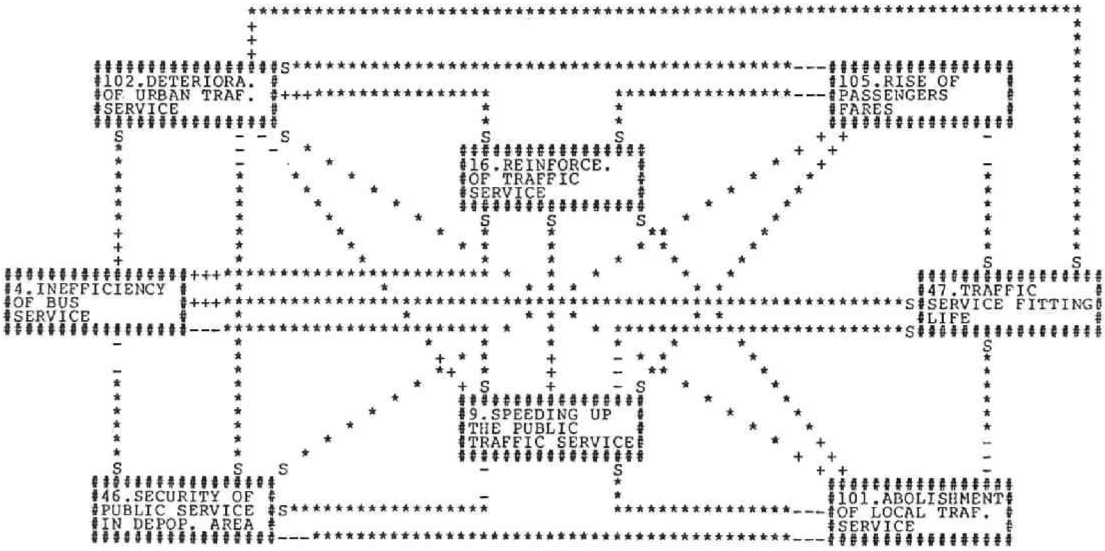
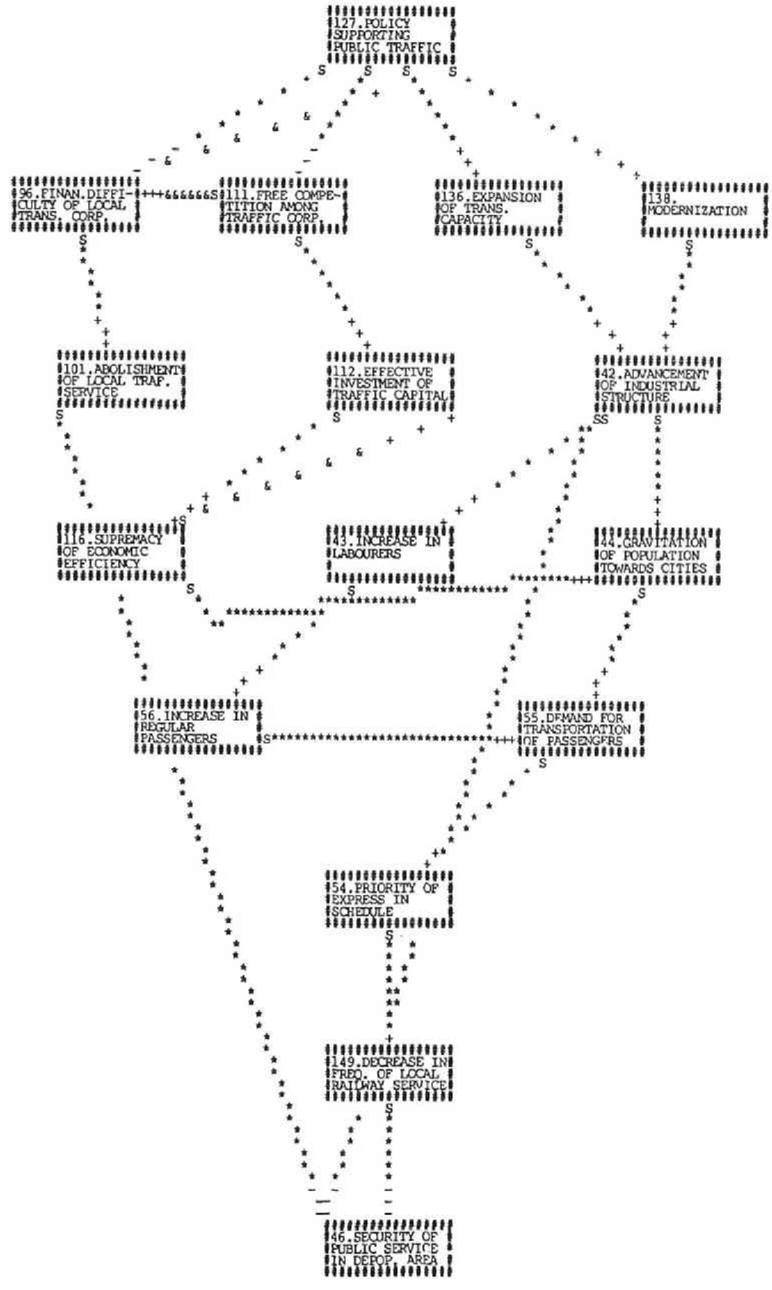


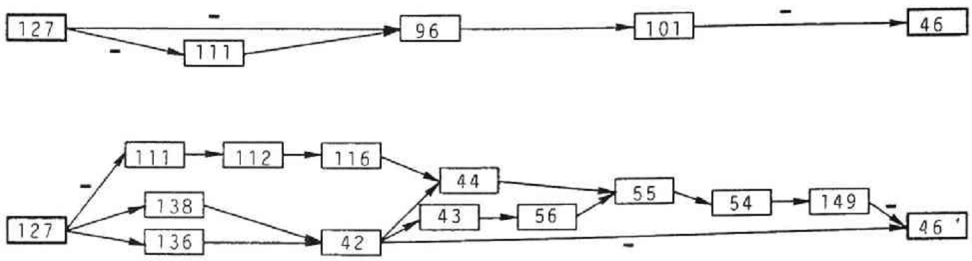
Fig. 4.4 Skeleton map relevant to key words SERVICE and FARE

DO YOU WANT TO KNOW DETAILED CAUSAL STRUCTURE OF ANY RELATION?
:
YES
INPUT STARTING NODE
:
127
INPUT ENDING NODE
:
46



(a)

(continue to next page)



(b)

Fig. 4.5 Causal effects of policy supporting public traffic (Concept 127) on security of public service in depopulated area (Concept 46).
(a) Hierarchical graph from Concept 127 to Concept 46.
(b) Two causal structures composing hierarchical graph.

population towards cities (44)," which have negative effects on "the security of public service in depopulated area (46)." Consequently, as far as the problem of security of public service in the depopulated area is concerned, only such policies should be taken as to ease the financial difficulties (96) and control the free competition (111) under fundamental policies controlling the advancement of industrial structure and the gravitation of population.

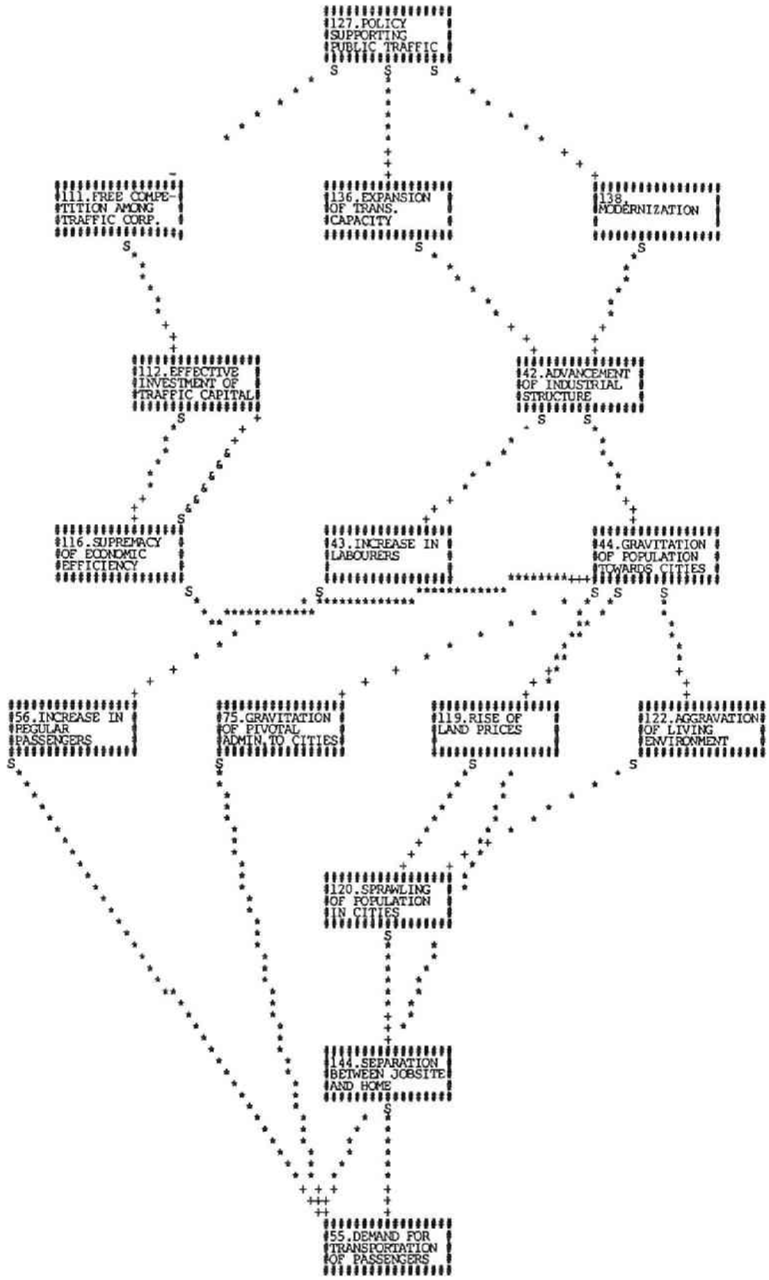
Example 3:

Suppose that referring to the skeleton map of Fig.4.1, the decisionmaker focuses on a serious problem of decrease in demands for passengers, i.e., "demands for transportation of passengers (55)" and investigates how the "(national) policy supporting public traffic (127)," which is a significant Concept 127, affects the Concept 55 in Fig.4.1. He enters the numbers of those two concepts, 127 and 55, and the system prints out a hierarchical graph from Concept 127 to Concept 55 shown in Fig.4.6(a). The graph is found to consist of two causal structures (i) and (ii) shown in Fig.4.6(b). The former structure (i) indicates positive

DO YOU WANT TO KNOW DETAILED CAUSAL STRUCTURE OF ANY RELATION?
:
YES

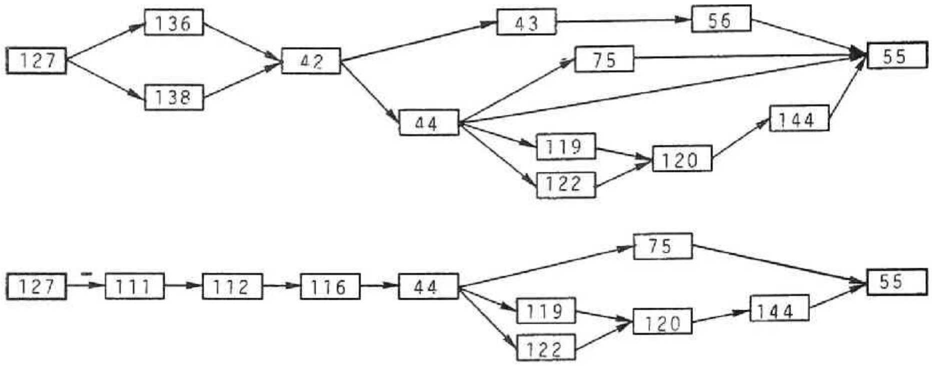
INPUT STARTING NODE
:
127

INPUT ENDING NODE
:
55



(a)

(continue to next page)



(b)

Fig. 4.6 Causal effects of policy supporting public traffic (Concept 127) on demand for transportation of passengers (Concept 55).

effects of the "policy supporting public traffic (127)" on "demands for transportation of passengers (55)" from the viewpoint of industrial structures. That is, the policy promotes "expansion of transportation capacity (136)" and "modernization (138)," and then causes "the advancement of industrial structure (42)," which promotes "the gravitation of population towards cities (44)" and "increase in labourers (43)" leading to the increase in "demand for transportation of passengers (55)." The latter structure (ii) is a causal network carrying negative effects of the policy. The structure shows that "free competition among traffic corp. (111)," which promotes "gravitation of population towards cities (44)" leading to the increase in "demand for transportation of passengers (55)" through such economic aspects as "effective investment of traffic capital (112)" and "supremacy of economic efficiency (116)," and policy (127) suppresses this Concept 111.

Consequently, as for the problem to increase the demands for transportation of passengers, we can know that the policy supporting public traffic may be indeed effective as far as aspects of the industrial structures are concerned, but at the same time that its unfavourable side-effects on economic aspects may be possible.

As above, the decisionmaker can focus his subject domain and form his overall and detailed understanding of the problem through the alternative process of interactive retrieval with the knowledge base.

4.3 Considerations on Universal Relations and Quantification of Causal Effects by Discriminating Positive and Negative Components

4.3.1 Consideration on Universal Relations in Causal Structures

In cognitive maps concerning societal problems characterized as having so many aspects such as political and economic ones, the causal structure between two concepts is not simple at all but consists of many sequences, each of which represents different aspects of the problem. The evaluated causal relationship between such two concepts according to the mathematical operations shown is hardly purely plus nor purely minus but proves to have both effects coincidentally, i.e., *universal* (u). This is because in the cognitive maps as a result of the integration of several experts' opinion from their own viewpoints the difference of aspects of their problem recognition appears. Even in an expert's recognition, when he notices side issues besides the main problems as a result of detailed analysis, such an apparently contradictory evaluation in which plus and minus relationships coexist can be made. Moreover, in societal problems it is quite often that we find a causal structure where an event causes another event which turns out to be a cause of the original event, that is, a closed loop structure, and if a sign of such a loop is minus, the evaluation through it will be necessarily universal.

As Fig.4.5(b) denotes, the graph is found to consist of two causal structures (i) and (ii). The former structure indicates positive effects of the "policy supporting public traffic (127)" on "security of public service in depopulated area (46)" by easing "financial difficulties of local transportation corporation (96)" and "controlling free competition among traffic corporations (111)." The latter structure (ii) is a causal network mainly carrying negative effects of the policy and includes two pivotal concepts "advancement of industrial structure (42)" and "gravitation of population towards cities (44)."[2] Comparing those structures with each other, we can find that the latter structure involves a larger variety of concepts and more sequences, while the former structure consists of just two sequences being made of four concepts. This seems to suggest that from the major points of view, the policy suppresses the security of public service in depopulated area rather than contributes to it even though the evaluation determined according to the operations turns out to be just universal, that is, it can be considered that the effect on the concept (46) carried by the concept (127) is essentially negative instead of universal.

This example demonstrates the inherent limitation of cognitive maps in treating configurations of evaluative relationships. Namely, however minor a belief it may be and even though it captured just a partial aspect of the whole structure, once it is included in the causal structures and when its evaluation conflicts with the rest, the resulting evaluation as a whole

[2] In this structure, a substructure from (127) to itself (127->111->96->127) represents that for the local traffic the easing(-) of "free competition (111)" due to the policy (127) resolves "financial difficulty of local transportation corporation (96)," which makes the policy (127) meaningless. This negative feedback loop inevitably makes such a sequence involving it as the one of 127->111->112->116->44->55->54->149->46 a universal one.

comes to be universal, and there is no way for such further evaluation as to which component of the effect is predominant. When we utilize such cognitive maps in DSS to help decisionmakers rank alternatives according to their goals and preference characteristics, however, instead of the traditional purely qualitative way of evaluating indirect causal relationships on cognitive maps, some alternative and extended methodology will be required that enables flexible evaluation, preserving the structural properties cognitive maps originally have.

4.3.2 Definition of Positive and Negative Causal Effects through Sequences

For this purpose, a method for separate quantification of those dual effects of a cause concept on an effect concept in signed diagraphs is to be developed, which takes into account that the effects are weakened in propagating through sequential intermediate causal relations and are strengthened in integrating a number of sequential effects existing in parallel.

In this thesis in order to develop such a method, we introduce the following three assumptions:

(A-1) Coexistence of various degrees of positive and negative effects are allowed in the same relationship.

(A-2) An effect of a concept on another concept through a sequence is weakened according to its length, that is, the number of causal assertions of the sequence.

(A-3) The more sequences exist in parallel, the more strengthened an effect from a concept on another one is.

The first assumption is based on our interpretation of the universal relationship as described in the previous section (Cartwright and Harary, 1970). The second assumption reflects the fact that the more intensity with which the author recognizes a causal relationship between two concepts, the less the number of intermediary concepts along the sequence from a cause concept to an effect concept would be, namely, the more the author would exclude the explanatory concepts for the effect concept making direct expressions. The third assumption would also be reasonable considering that the more aspects of the societal problems the cause concept exerts influences on, the more the author makes repeated descriptions in his documents from various viewpoints, which makes the total effect become greater.

Definition 4.1

Let $E_{ij}^{(m)}$ and $I_{ij}^{(m)}$ be numbers of positive and negative sequences of length m contained in a particular causal relationship from concept i to j, respectively. From assumptions **(A-2)** and **(A-3)**, positive and negative effects of concept i on concept j, $\tilde{p}_{ij}$ and $\tilde{n}_{ij}$ can be naturally defined as follows, respectively.

$$\begin{aligned} \tilde{p}_{ij} &= \sum_{m=1}^{\infty} f(m) E_{ij}^{(m)} && \text{: positive effect} \\ \tilde{n}_{ij} &= \sum_{m=1}^{\infty} f(m) I_{ij}^{(m)} && \text{: negative effect,} \end{aligned} \tag{4-1}$$

where f(m) is a monotonous decreasing function of length m specifying attenuation characteristics of causal effect in propagating through a sequence.

When we choose as f(m) the following monotonous decreasing and differentiable function

$$f(m) = z^m \quad (0 < z < 1), \tag{4-2}$$

where z is a coefficient determining the degree of attenuation, then the symmetrical and distributive law on calculation of the causal effects through relaying concepts in a causal structure is held as follows.

Theorem 4.1

Let j denote an intermediary concept laid on a sequence connecting concept i with concept k. The components of positive and negative effects of concept i on concept k through concept j, $\tilde{p}_{ik(j)}$ and $\tilde{n}_{ik(j)}$, are calculated as follows, respectively.

$$\begin{aligned} \tilde{p}_{ik(j)} &= \tilde{p}_{ij}\tilde{p}_{jk} + \tilde{n}_{ij}\tilde{n}_{jk} \\ \tilde{n}_{ik(j)} &= \tilde{p}_{ij}\tilde{n}_{jk} + \tilde{n}_{ij}\tilde{p}_{jk} \end{aligned} \qquad (4\text{-}3)$$

(proof)

From the definition of signs of sequences, numbers of positive and negative sequences connecting concept i with concept k through concept j, $E_{ik(j)}$ and $I_{ik(j)}$, are given by

$$\begin{aligned} E_{ik(j)} &= E_{ij}E_{jk} + I_{ij}I_{jk} \\ I_{ik(j)} &= E_{ij}I_{jk} + I_{ij}E_{jk}, \end{aligned} \qquad (4\text{-}4)$$

respectively (Burns and Winstead, 1982). For simplicity, assuming that lengths of all sequences connecting i with j and j with k are equal to m and m', respectively, the components of causal effects of concept i on concept k through concept j are represented by

$$\begin{aligned} \tilde{p}_{ik(j)} &= f(m+m')E_{ik(j)}^{(m+m')} = f(m+m')E_{ik(j)} \\ \tilde{n}_{ik(j)} &= f(m+m')I_{ik(j)}^{(m+m')} = f(m+m')I_{ik(j)}. \end{aligned} \qquad (4\text{-}5)$$

Equations (4-3) are direct consequences of replacing (4-2), (4-4) into (4-

5) and referring to (4-1). *Q.E.D.*

Based on **Theorem 4.1**, a set of matrix operations can be derived, which contributes to an efficient computation of the causal effects for given causal structures.

Definition 4.2

Let P and N be two binary matrices denoting existence of positive and negative causal assertions (direct plus and minus relations) between concepts, that is, positive and negative valency matrices,

$$\mathbf{P} = [p_{ij}]$$
$$\mathbf{N} = [n_{ij}],$$

respectively, where $p_{ij}(n_{ij})$ equals to 1 when there exists a direct plus (minus) causal relationship between any pair of concepts (i,j) and equals to 0 otherwise.

Theorem 4.2

Let $\tilde{P}^{(m)} = [\tilde{p}_{ij}^{(m)}]$ and $\tilde{N}^{(m)}=[\tilde{n}_{ij}^{(m)}]$ denote matrices whose elements are positive and negative causal effects through the sequences of length m, respectively. Then,

$$\begin{aligned} \tilde{\mathbf{P}}^{(m)} &= \tilde{\mathbf{P}}^{(m-1)}\ \tilde{\mathbf{P}}^{(1)} + \tilde{\mathbf{N}}^{(m-1)}\ \tilde{\mathbf{N}}^{(1)} \\ \tilde{\mathbf{N}}^{(m)} &= \tilde{\mathbf{P}}^{(m-1)}\ \tilde{\mathbf{N}}^{(1)} + \tilde{\mathbf{N}}^{(m-1)}\ \tilde{\mathbf{P}}^{(1)}, \end{aligned} \tag{4-6}$$

where $\tilde{P}^{(1)} = [\tilde{p}_{ij}^{(1)}] = [f(1)p_{ij}] = z\mathbf{P}$ and $\tilde{N}^{(1)} = [\tilde{n}_{ij}^{(1)}] = [f(1)n_{ij}] = z\mathbf{N}$.

Lemma 4.1

$\widetilde{\mathbf{P}}^{(m)}$ and $\widetilde{\mathbf{N}}^{(m)}$ can be expressed by $\widetilde{\mathbf{P}}^{(1)}$ and $\widetilde{\mathbf{N}}^{(1)}$ as follows:

$$\begin{aligned}\widetilde{\mathbf{P}}^{(m)} &= [(\widetilde{\mathbf{P}}^{(1)} + \widetilde{\mathbf{N}}^{(1)})^m + (\widetilde{\mathbf{P}}^{(1)} - \widetilde{\mathbf{N}}^{(1)})^m]/2 \\ \widetilde{\mathbf{N}}^{(m)} &= [(\widetilde{\mathbf{P}}^{(1)} + \widetilde{\mathbf{N}}^{(1)})^m - (\widetilde{\mathbf{P}}^{(1)} - \widetilde{\mathbf{N}}^{(1)})^m]/2\end{aligned} \tag{4-7}$$

Let denote the matrices of positive total causal effect and negative one by $\widetilde{\mathbf{P}} = [\widetilde{p}_{ij}]$ and $\widetilde{\mathbf{N}} = [\widetilde{n}_{ij}]$, respectively. $\widetilde{p}_{ij}$ represents the positive total causal effect from concept i to concept j through all sequences with various lengths and similarly for $\widetilde{n}_{ij}$.

Theorem 4.3

The positive and negative total causal effect matrices $\widetilde{\mathbf{P}} = \sum_{l=1}^{\infty} \widetilde{p}^{(l)}$ and $\widetilde{\mathbf{N}} = \sum_{l=1}^{\infty} \widetilde{n}^{(l)}$ can be represented by $\widetilde{\mathbf{P}}^{(1)}$ and $\widetilde{\mathbf{N}}^{(1)}$ as

$$\begin{aligned}\widetilde{\mathbf{P}} &= [(\widetilde{\mathbf{P}}^{(1)} + \widetilde{\mathbf{N}}^{(1)})[\mathbf{I} - (\widetilde{\mathbf{P}}^{(1)} + \widetilde{\mathbf{N}}^{(1)}]^{-1} \\ &\quad + (\widetilde{\mathbf{P}}^{(1)} - \widetilde{\mathbf{N}}^{(1)})[\mathbf{I} - (\widetilde{\mathbf{P}}^{(1)} + \widetilde{\mathbf{N}}^{(1)}]^{-1}]/2 \\ \widetilde{\mathbf{N}} &= [(\widetilde{\mathbf{P}}^{(1)} + \widetilde{\mathbf{N}}^{(1)})[\mathbf{I} - (\widetilde{\mathbf{P}}^{(1)} + \widetilde{\mathbf{N}}^{(1)}]^{-1} \\ &\quad - (\widetilde{\mathbf{P}}^{(1)} - \widetilde{\mathbf{N}}^{(1)})[\mathbf{I} - (\widetilde{\mathbf{P}}^{(1)} + \widetilde{\mathbf{N}}^{(1)}]^{-1}]/2,\end{aligned} \tag{4-8}$$

where I represents an unit matrix.

In Equation (4-2), parameter z specifies the degree to which causal effects are attenuated in propagating through sequences. As shown in that equation, the value of z is restricted any positive real numbers not exceeding unity. Since f(m) is a monotonous decreasing function, the following theorem holds, which restricts more the value of z, that is,

Theorem 4.4

Unless all the eigenvalues of a matrix $(\widetilde{\mathbf{P}}^{(1)} + \widetilde{\mathbf{N}}^{(1)})$ exceed 1, then the matrices $\widetilde{\mathbf{P}}$ and $\widetilde{\mathbf{N}}$ converge into finite value matrices.

(proof)

From **Lemma 4.1**, the total causal effect matrices $\widetilde{\mathrm{P}}$ and $\widetilde{\mathrm{N}}$ are represented as follows, respectively.

$$\begin{aligned} \widetilde{\mathrm{P}} &= \Big(\sum_{l=1}^{\infty}(\widetilde{\mathbf{P}}^{(1)} + \widetilde{\mathbf{N}}^{(1)})^{l} + \sum_{l=1}^{\infty}(\widetilde{\mathbf{P}}^{(1)} - \widetilde{\mathbf{N}}^{(1)})^{l}\Big)/2 \\ \widetilde{\mathbf{N}} &= \Big(\sum_{l=1}^{\infty}(\widetilde{\mathbf{P}}^{(1)} + \widetilde{\mathbf{N}}^{(1)})^{l} - \sum_{l=1}^{\infty}(\widetilde{\mathbf{P}}^{(1)} - \widetilde{\mathbf{N}}^{(1)})^{l}\Big)/2 \end{aligned} \qquad (4\text{-}9)$$

By the well-known convergence test for the vector power series[3], $\sum_{l=1}^{\infty}(\widetilde{\mathbf{P}}^{(1)} + \widetilde{\mathbf{N}}^{(1)})^{l}$ and $\sum_{l=1}^{\infty}(\widetilde{\mathbf{P}}^{(1)} - \widetilde{\mathbf{N}}^{(1)})^{l}$ converge absolutely if all the eigenvalues of both $(\widetilde{\mathrm{P}}^{(1)} + \widetilde{\mathbf{N}}^{(1)})$ and $(\widetilde{\mathrm{P}}^{(1)} - \widetilde{\mathbf{N}}^{(1)})$ do not exceed unity. In general, let **A** and **B** denote two real valued positive definite matrices, and u and u' be the eigenvalues of **(A+B)** and **(A-B)** having the maximum absolute values, respectively. Then, there exists an inequality such that $u \geqq |u'|$. From the inequality and Equation (4-9), we can say that the fact that eigenvalues of $(\widetilde{\mathbf{P}}^{(1)} + \widetilde{\mathbf{N}}^{(1)})$ do not exceed unity is sufficient for both $\bar{\mathrm{P}}$ and $\bar{\mathrm{N}}$ to converge. *Q.E.D.*

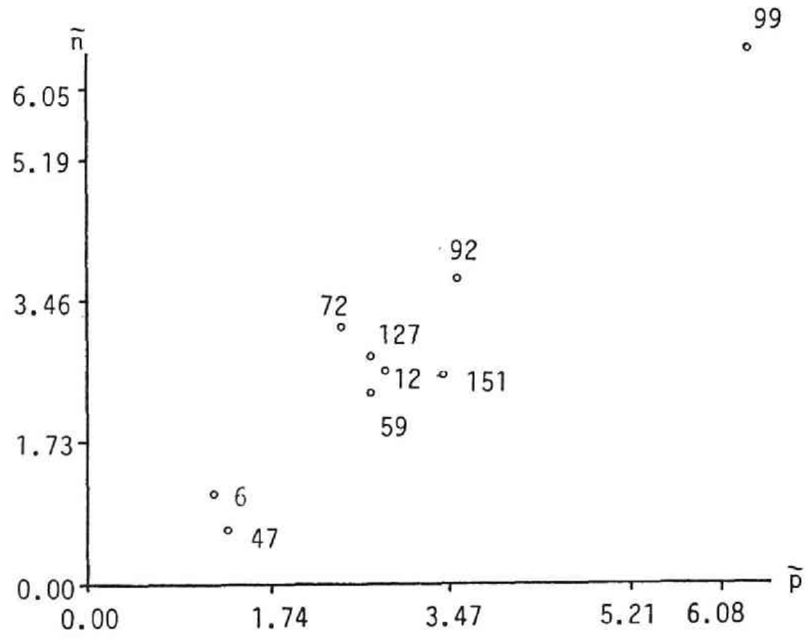
Therefore, letting λ_{max} be the maximum eigenvalue of the summed matrix of positive and negative valency matrices, $(\mathbf{P} + \mathrm{N})$, the coefficient z can be determined arbitrarily in the region of

[3] A matrix power series $\sum_{n=0}^{\infty} a_n \mathbf{X}^n$ converges absolutely if all the eigenvalues of X exist in the interior of the circle of convergence of the power series $\sum_{n=0}^{\infty} a_n \mathbf{X}^n$, and diverges if any of them exists outside of the circle.

$0 < z < 1/\lambda_{max}$ ($\lambda_{max} > 1$) or $0 < z < 1$ ($\lambda_{max} \leqq 1$), since $\widetilde{\mathbf{P}}^{(1)} + \widetilde{\mathbf{N}}^{(1)}$ equals to $z(\mathbf{P}+\mathbf{N})$.

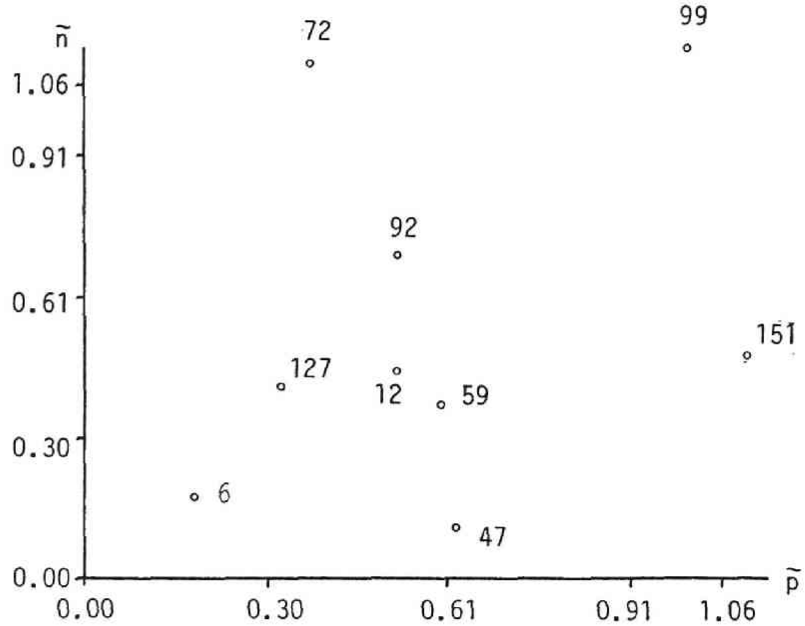
In general, there exist so many sequences with different length m from a cause concept i to an effect concept j. As shown in Equation (4-2), the coefficient z ($0 < z < 1$) specifies the degree to which the causal effect through sequences is attenuated, and the effect through a sequence of length m is calculated as z^m. Therefore, determining the coefficient z as a larger value, namely considering the degree to which the causal effect is weakened in propagating through long sequences less, means that causal effects on a goal concept in even an indirect way through long sequences in cognitive maps are also taken into consideration, while the smaller coefficient z consequently represents that such an indirect effect can be neglected. Let us illustrate how the evaluation of each policy concept on $\widetilde{\mathbf{P}} \times \widetilde{\mathbf{N}}$ varies with the variation of the value of z using the concrete example extracted from our constructing knowledge base.

Fig.4.7(a)–(c) show the plotted locations of policy concepts 6, 12, 47, 59, 72, 92, 99, 127, and 151 on the two-dimensional $\widetilde{\mathbf{P}} \times \widetilde{\mathbf{N}}$ space with three different values of z = 0.52, 0.49, and 0.42, respectively. Those policy concepts exert a goal concept "aggravation of traffic jam (103)" through a causal structure consisting of 80 concepts represented by the valency matrix shown in Fig.4.7(d) and $z_{max} = 1/\lambda_{max}$ is calculated as 0.524 from the valency matrix according to the theory mentioned before in this section. As these figures illustrate, with a large value of z nearly equal to z_{max}, almost all policy concepts are located along the line of $\widetilde{p} = \widetilde{n}$, which represents an evaluation that neither component of the effect is not negligible making it difficult to decide whether those policies are favourable or unfavourable to attain the goal, that is, to suppress the



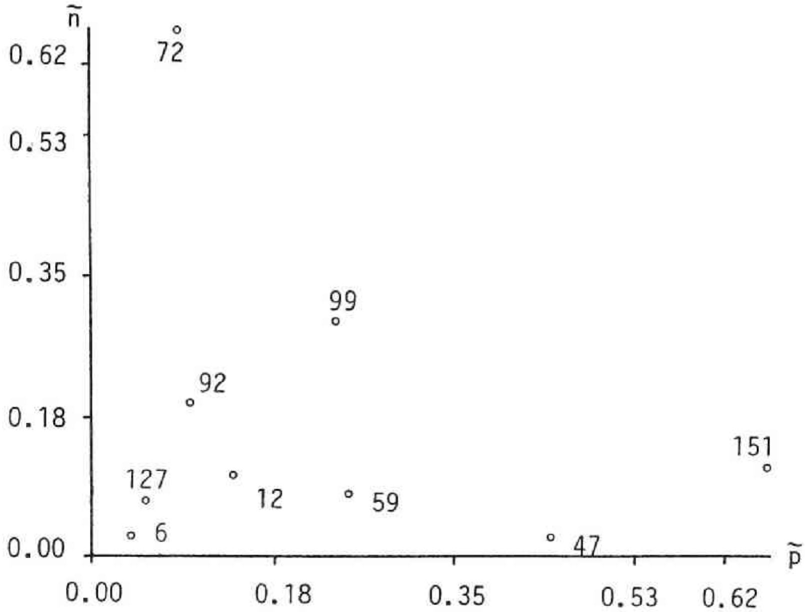
ñ
6.05
5.19
3.46
1.73
0.00
0.00
1.74
3.47
5.21
6.08
p̃
99
92
72
127
12
151
59
6
47

(a)



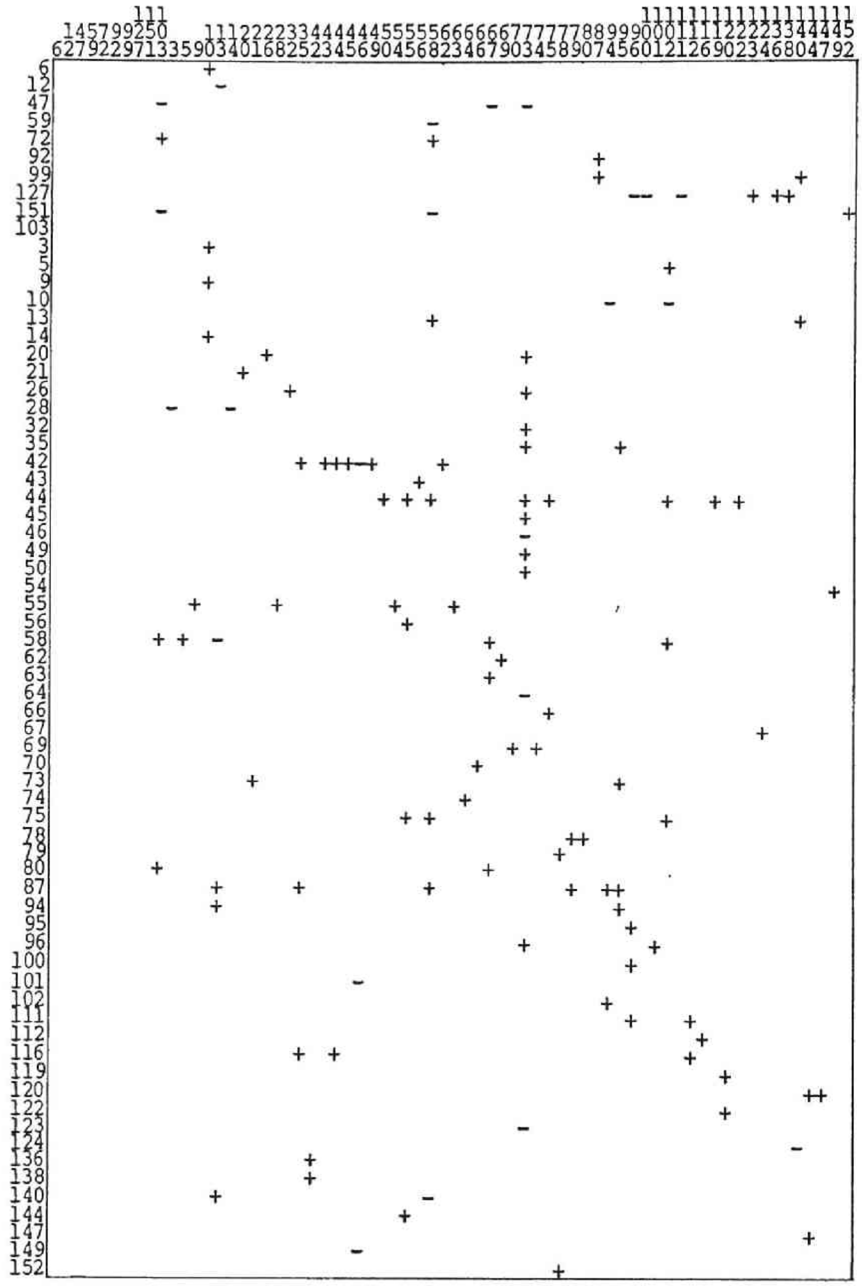
ñ
1.06
0.91
0.61
0.30
0.00
0.00
0.30
0.61
0.91
1.06
p̃
72
99
92
151
127
12
59
6
47

(b)



(c)

Fig. 4.7 Causal effects of nine policies on suppressing aggravation of traffic pollution(Concept 103) plotted on P x N space with three different coefficients of z. (a) z= 0.52. (b) z=0.49. (c) z=0.42. (d) Valency matrix of a causal structure from the policies to Concept 103. Policies are: improvement of traffic terminals (Concept 6), self-imposed control of car usage (12), traffic service fitting life (47), control of parking cars (59), unfair priority system for car-product (72), unfair traffic policy for cars (92), partial public investment in roads (99), policy supporting public traffic (127), and setting no truck passing area in daytime (151).



(d)

(continued)

concept of 103. As the value of z gets less, each location departs from the line of $\tilde{p} = \tilde{n}$ and approaches to either $\tilde{p}$ axis or $\tilde{n}$ axis, that is, the distinction whether each policy contributes towards the attainment of the goal concept or not becomes clearer, in a general trend.

This reflects the fact that determining coefficient z as a larger value denotes the decisionmaker's attitude not to miss the effects through long sequences in cognitive maps and to evaluate comprehensively the accumulated causal effects from so many aspects or viewpoints including even those indirect, that is, long-sequenced effects, which makes the evaluation of policies *neutral*, that is, the evaluation far from purely positive and purely negative. On the contrary, determining z as a smaller value reflects the decisionmaker's attitude to expect causal effects through shorter sequences more, restricting his view to the more direct relationships and neglecting the indirect ones through long sequences. As a result of that, the more distinctive evaluation on the favourability of policies can be made. Therefore, the determination of coefficient z can be a trade-off problem for the decisionmaker between the range of view required to the evaluation of the policies and the distinctiveness of those evaluations, which depends on the decisionmaker's individual attitude towards his goal.

Assuming that there exist n sequences with a length m between a policy concept i and a goal concept j, in order for the measure of effect of concept i on concept j to be equal to the one of the causal relationship where two concepts are linked directly, n is determined as $z^{-(m-1)}$. That is, corresponding to the length m of a sequence for which a decisionmaker evaluates the accumulation of causal effects through sequences, the number of sequences, n, can be calculated. A decisionmaker, comparing the causal structure determined by the designated length m and the calculated value of n with intuitively-comprehensible direct causal effect and considering the trade-off between a range of view and distinctiveness, a decisionmaker appoints a value of z which he considers appropriate.

4.4 Concluding Remarks

For problem solving or decisionmaking for such recent highly complex problems as societal and environmental problems requiring global understanding of the problem structure from many aspects, the necessity for the introduction of a decision support system which leads a decisionmaker to his final stage of decision process flexibly through the conversation with a knowledge base, i.e., a great amount of expertise specific to the problem domain, is increasing. In this chapter, based on such requirement, we took notice of a knowledge processing level of integrating a huge amount of expertise in advance to the final stage of choice that has been the major interest of the previous DSS's based on numerical models. Under such a motivation, we discussed on a DSS which helps a decisionmaker understand the problem in the stage of information gathering towards sound decisionmaking.

For this purpose, for the knowledge base using cognitive maps, we developed a knowledge retrieval system which helps a decisionmaker to grasp the global problem structure according to his various viewpoints. Next, in societal complexes, a causal relationship such as the one between a policy and a goal, i.e., a sign of causal sequences between concepts, can be rarely determined as either positive or negative, but usually consists of both kinds of effects, that is, a policy promotes the attainment of a goal as well as suppressing it. Therefore, we developed a method to separately quantify each kind of causal effects digraph-theoretically, considering that causal effects of policy candidates are weakened in propagating through causal sequences and strengthened in integrating effects along a number of causal sequences existing in parallel.

CHAPTER 5

IDENTIFICATION OF PROMININT POLICY ALTERNATIVES REFLECTING DECISIONMAKER'S CONFORMITY

5.1 Introduction

Using kinds of retrieval modes for the experts' views as causal assertions described in the previous chapter, it becomes possible to understand the problem by explicating the interwoven relationships among the elements forming the problem, as well as to clarify what opinions the experts have on that matter, and in what ways they are conflicting one another. Here, it comes to be possible for the decisionmaker to estimate experts' knowledge and to integrate his own views or policies into that. In general, such soft knowledge represented as documented statements offering different interpretations, and the knowledge-based DSS must be adaptable and extendable to meet the evolving goals of the user, that is, the decisionmaker. Therefore, it is required of DSS to contain any formal consideration of the subjective judgment and complex value structures, which proves to be especially useful for assisting sound decisions in the face of uncertainties and conflicting opinions.

In this chapter, using those measures quantifying to what degree some policy gives favourable effects as well as unfavourable ones, we propose a method to integrate experts' knowledge stored in the knowledge base with decisionmakers' evolving strategic policies for their decision problems.

Then, as an application of those ideas, we develop a knowledge-based DSS in which an alternative set is generated on the user's request, and the policy contributing best to the user's own utility is specified through an interactive conversation between the user and the system.

For this purpose, based on the measures quantifying the universal relations, we discuss the user's preference characteristics on the universal relation described in the previous chapter, and discuss a technique for interactive identification of the user's value function on the two attributes, *policy impact* and *policy consonance* (cf. Sawaragi, Iwai and Katai, 1984, 1985b, 1986a, 1986b).

5.2 Policy Validity Evaluation Function Reflecting Decisionmaker's Conformity

5.2.1 Measures for Extracting Decisionmaker's Conformity

The foregoing quantification method allows us to represent the effect of a policy alternative (a policy or a set of policy concepts) on a goal concept as a point in the two dimensional space having positive and negative axes. Decisionmaker's evaluation on the validity of the location of the point is determined by judging how far the point lies from the origin of the space and how close to the positive axis relative to the negative one. We introduce the following two attributes s_{ij} and c_{ij} to quantify the location of the points.

Definition 5.1

$$\begin{aligned} s_{ij} &= \tilde{p}_{ij} + \tilde{n}_{ij} \\ c_{ij} &= (\tilde{p}_{ij} - \tilde{n}_{ij})/(\tilde{p}_{ij} + \tilde{n}_{ij}) \end{aligned} \tag{5-1}$$

The attribute s_{ij} is a measure of how far the point representing the effect of policy alternative i on j lies from the origin of the space, and it can be interpreted as representing an impact of i on j as a whole, in other words, a "controllability" showing to what degree the effect is supposed to be observed in concept j as a result of taking i apart from whether it is positive or negative. We call this attribute *policy impact*. The attribute c_{ij} is a measure denoting the difference of a proportion of positive effect to the total effect and the one of negative effect to it, that is, a measure showing to what extent the positive effect dominates the negative one. c_{ij} can be considered as a measure denoting a kind of a "cognitive consonancy" of the effect of concept i on j. We call this *policy consonace*.

Investigating the location of the point, decisionmakers interpret the validity of the policy alternative corresponding to the point according to their own conformities. Decisionmakers' conformities depends on their standpoints, their noticing urgency of the problem, or the degree of *dissonance*, that is, a kind of pressure or apprehension they feel after taking some policy action due to the negative effect contained in that policy (Festinger, 1957). Therefore, the interpretations depend conspicuously on decisionmakers' conformities, for instance, in Fig.5.1, risk-evading decisionmakers or ones fearing the negative influence caused by their taking some policy will prefer (47) having larger c value; c = 0.71 to (151) having smaller c value; c = 0.40. On the contrary, in the case of aggressive and optimistic decisionmakers, (151) having larger s value; s = 1.57, will be recommended for its large controllability as a powerful policy and (6) having smaller s value; s = 0.33, will be discarded as blameless but less controllable.

In this way, based on their individual interpretations, decisionmakers

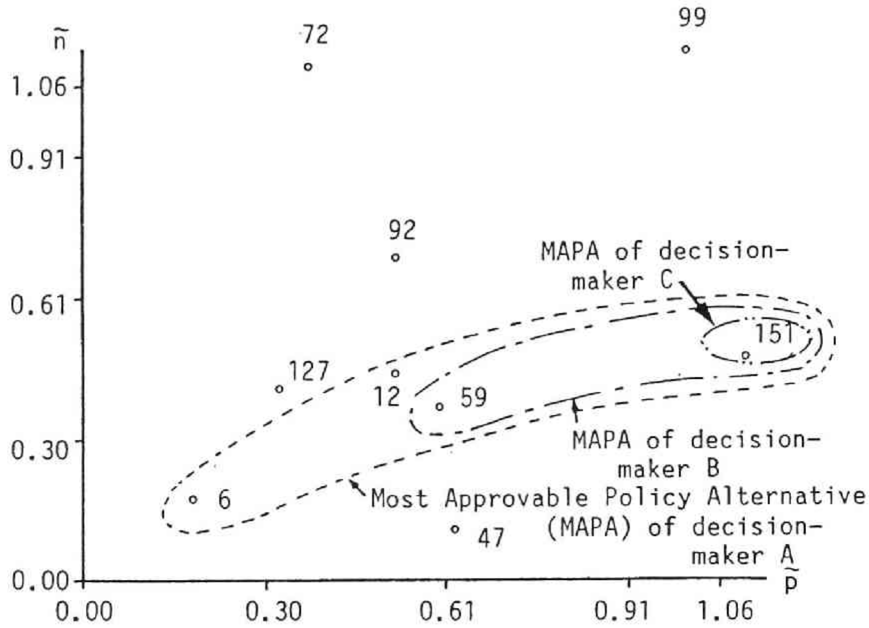


Fig. 5.1 Causal effects of nine policies on suppressing aggravation of traffic pollution (Concept 103) plotted on P x N space z=0.42. Plotted policies are the same ones with Fig. 4.7.

decide whether a policy alternative should be adopted or not. For the purpose of supporting such a process of decisionmaking, a technique is required such that it is capable of reflecting each decisionmaker's conformities in interpreting the validity of policy described by the two attributes s and c. Let $v(s_{ij}, c_{ij})$ be a quantification of a decisionmaker's evaluation for the validity of policy i on goal concept j. We call v(s,c) as *Policy Validity Evaluation Function*[1]. For some given goal concept, a policy which has more impact and more purely positive effects on it is more appropriate as a policy to be adopted. Namely, a policy with larger value of s and a value of c nearer to 1 is desirable. However, not in a simply-structured problem but in a societal complex one where a policy gives effects on so many aspects of the problem, a policy having a large value of s may have not only contributive (positive) causal sequences but also suppressing (negative) sequences as well to the goal concept, which makes the value of c decrease to 0 or even below 0. This means that the determination of v(s,c) is dealt with as a decisionmaker's trade-off problem between a policy's attractiveness on its controllability and misgivings about the interference of negative effects.

5.2.2 Policy Validity Evaluation Function and Preference Characteristics

If there are n policy concepts which can give effect on goal concept j appointed by a decisionmaker, the number of policies or policy sets, that is, policy alternatives whose validities should be evaluated is 2^n.[2] The choice of the ideal policy alternative among those 2^n alternatives depends

---

[1] It is possible to construct this function using $\tilde{p}$ and $\tilde{n}$ of Eq.(4-1) in place of s and c of Eq.(5-1). However, as for the extraction of the preference characteristics from the user, i.e., the decisionmaker, it is difficult to do that by asking him the preference on positive effect $\tilde{P}$ and negative effect $\tilde{\mathbf{N}}$ individually since the preference evaluation on one attribute depends on the evaluation on the other one. If we are to characterize the user's preference using these two attributes S and C instead of $\tilde{P}$ and $\tilde{N}$, the user would be asked, for instance, how intense an effect is required, or to what degree the negative effect can be allowed against the positive one, both of which seem easy for the user to judge and answer subjectively.

[2] We calculate the positive and negative components of an aggregate effect for a policy alternative, i.e., a set of policy concepts by summing each component for individual policy concepts respectively.

on the decisionmaker's conformity as described above. Let (s_{ij}, c_{ij}) and $(s_{i'j}, c_{i'j})$ be the quantification of effects of two policy alternatives i and i', respectively. Then, the validity evaluation function v(s,c) is a scalar function which satisfies the following equation:

$$v(s_{ij},c_{ij}) \geqq v(s_{i'j},c_{i'j}) \Longleftrightarrow a_i \gtrsim a_{i'}, \tag{5-2}$$

where the right hand side formula means policy i is preferable or indifferent to policy i'. The function $v(s_{ij}, c_{ij})$ can be characterized generally as follows. In case $c_{ij}>0$, that is, the positive effect exceeds the negative one, the larger the absolute value of s_{ij} and the nearer to 1 the value of c_{ij} is, the greater the validity of policy i for goal concept j is. When the negative effect is predominant, on the contrary, policy concept i having a larger absolute value of s_{ij} and a value of c_{ij} nearer to -1 is less appropriate for goal concept j[3].

Concerning the boundary $c_{ij} = 0$, we make the following assumption:

(A-4) In case of $c_{ij}=0$, $v(s_{ij}, c_{ij})$ is assumed to be 0 for any values of s_{ij}.

This assumption reflects the psychological views on human decisionmakers suggested by Festinger under the name of *theory of cognitive dissonance* (Festinger, 1957). Cognitive dissonance is a noxious state or a postdecision regret, which humans feel as a kind of pressure, tension, or inconsistency as a result of some decision. Since decision involves a

---

[3] Note that the above characteristics in each case of $c > 0$ and $c < 0$ is reversed when the content of the goal concept is not the one to be attained but to be suppressed, i.e., a policy suppressing more the goal concept is considered as a desirable one. In the rest of this paper, we only deal with the former case, that is, concept i is a goal to be attained.

selection of one among a set of alternatives, it necessarily entails forsaking the attractive features of rejected alternatives and accepting the negative features of the selected alternatives. The severity or intensity of cognitive dissonance varies with how important the cognitions involved are and how close the magnitudes of attractiveness of the rejected alternative and of the negativity of the accepted one are mutually. In case the value c_{ij} is nearly equal to 0, if a decisionmaker were to adopt that policy set, he should have to accept the negative effect of the policy, while he should have to discard the attractive positive effect if he were not to adopt it. Thus, the duality of the policy effect is dissonant with each other, and a policy set i having a value of c_{ij} nearly equal to 0 and a great value of s_{ij} evokes a severe dissonant situation where a decisionmaker feels anxious about his decision. In our opinion, an efficient DSS should not bring about such a dissonant situation in consequence of the system-suggesting choice however large an impact it may have, so we assume $v(s_{ij}, c_{ij})$ of any policy sets with a value $c_{ij} = 0$ to be uniformly 0. Indeed, in such a situation the severity would be decreased according to the effect sensitivity s_{ij}, but at the same time the significance of i as a policy to solve the problem of goal concept j would be lost on the contrary, and the Policy Validity Evaluation Function can be thought of as null.

From the assumption **(A-1)** in the previous chapter, we can make the following assumption:

(A-5) v(s,c)'s on $\mathbf{S} \times \mathbf{C}_+$ $(s > 0,\ c > 0)$ and $\mathbf{S} \times \mathbf{C}_-$ $(s > 0,\ c < 0)$ are *measurable two-attribute value functions under certainty.*

Moreover, to simplify the extraction process, we assume as follows:

(A-6) Attributes S and $C_+ (C_-)$ are *weak difference independent* of each other, that is, the ordering of preference difference depends only on the values of one attribute and not on the fixed values of the other attribute.

Attribute S's weak difference independence of attribute C_+ is formulated as follows: Given any $w_S, x_S, y_S, z_S \in S$ and some $w_C \in C_+$,

$$(w_S, w_C)(x_S, w_C) \overset{*}{\gtrsim} (y_S, w_C)(z_S, w_C), \\ (w_S, x_C)(x_S, x_C) \overset{*}{\gtrsim} (y_S, x_C)(z_S, x_C) \tag{5-3}$$

for any $x_C \in C$, where we shall interpret $wx \overset{*}{\gtrsim} yz$ to mean that the strength of preference for w over x is greater than or equal to the strength of preference for y over z.

Based on assumptions **(A-5)** and **(A-6)**, and extension of the *utility theory under uncertainties* to *measurable value function under certainties* by Dyer and Sarin (Dyer & Sarin, 1979), v(s,c) can be decomposed into two measurable one attribute conditional value functions, $v_S(s)$ and $v_S(c)$, as follows[4]:

Theorem.5.1

If attributes S and C are mutually weak difference independent, then the two-attribute measurable value function can be written in the form

$$v(s,c) = k_S v_S(s) + k_C v_C(c) + (1-k_S-k_C) v_S(s) v_C(c), \tag{5-4}$$

where

[4] From the theory of Keeney et al., the utility function with multi-attributes can be decomposed into a multiplication of each single attribute functions under the assumption of weak difference independent (Keeney & Raiffa, 1976). For value functions, Dyer et al. showed that the same decomposition rule holds under the assumption (Dyer and Sarin, 1979).

1. $v(s,c)$ is normalized by $v(s^o, c^o) = 0$ and $v(s^*, c^*) = 1$ for arbitrary s^* and c^* such that $(s^*, c^o) > (s^o, c^o)$ and $(s^o, c^*) > (s^o, c^o)$.

2. $v_S(s)$ is a conditional measurable value function with the condition of $c = c^*$ on S normalized by $v_S(s^o) = 0$ and $v_S(s^*) = 1$.

3. $v_C(c)$ is a conditional measurable value function with the condition of $s = s^*$ on C normalized by $v_C(c^o) = 0$ and $v_C(c^*) = 1$.

4. $k_S = v(s^*, c^o)$.

5. $k_C = v(s^o, c^*)$.

(Proof)

Because of the assumption that attribute c is weak difference independent of attribute S, it follows that

$$v(s, c) = g(s) + h(s)v(s', c), \quad \text{for all c and s},$$

where h is always positive and s' is arbitrarily chosen specific amount of attribute s. Functions f and g, in general, will depend on the specific value of s' but not on the variable c.

Then, conditional value function $v_C(c;s)$ over attribute C with the fixed value of s can be defined as

$$v_C(c;s) = \frac{v(s, c) - v(s, c^o)}{v(s, c^*) - v(s, c^o)} \tag{5-5}$$

where $v_C(s, c^o) = 0$ and $v_C(s, c^*) = 1$ for all s, and c^o and c^* are the least preferred level and the most preferred level of attribute C. This means that the normalized conditional value function $v_C(c;s)$ over attribute C is the same for all s. So all conditional value functions over S can be represented by $v_C(c) = v_C(c;s^*)$. Considering this and Eq.(5-5), we can obtain

$$v(s,c) = v(s,c^o) + [v(s,c^*) - v(s,c^o)]v_C(c). \tag{5-6}$$

Similarly for a case attribute s is weak difference independent of attribute C, we can obtain

$$v(s,c) = v(s^o,c) + [v(s^*,c) - v(s^o,c)]v_S(s), \tag{5-7}$$

where $v_S(s) = v_S(s;c^*)$.

From Eq.(5-6) and what is yielded by substituting $c = c^o$ into Eq.(5-7), and rearranging, we can obtain

$$\begin{aligned} v(s,c) = {} & v(s^*,c^o)v_S(s) + v(s^o,c^*)v_C(c) \\ & + [1 - v(s^o,c^*) - v(s^*,c^o)]v_C(c)v_S(s), \end{aligned} \tag{5-8}$$

that is,

$$v(s,c) = k_C v_C(c) + k_S v_S(s) + (1 - k_C - k_S)v_C(c)v_S(s), \tag{5-9}$$

where $k_C = v(s^o,c^*)$ and $k_S = v(s^*,c^o)$. *Q.E.D.*

Based on assumption **(A-4)**, if we assume s^o and c^o to be nearly equal to 0, k_S and k_C can be regarded as approximately equal to 0. Then, equation (5-4) in the theorem can be expressed approximately as

$$v(s,c) = v_S(s)v_C(c). \tag{5-10}$$

This means that the validity evaluation function v(s,c) can be specified through a decisionmaker's two individual evaluations, that is, the one on policy impact or controllability $v_S(s)$ and on policy consonance $v_C(c)$, that is, "quantity" and "quality" of a specified policy alternative.

5.3 Identification of Prominent Alternatives Using Policy Validity Evaluation Function

5.3.1 User-Oriented Determination of Policy Validity Evaluation Function

It is natural that the decisionmakers' preference structure for the case of a policy alternative promoting a goal concept ($c > 0$) differs essentially with the case of suppressing ($c < 0$). Therefore, whenever a goal concept is selected, both the determination of $v_S(s)$ and $v_C(c)$ must be done by extracting the preference characteristics for the case of $c > 1$ and $c < 1$, separately. As far as the weak difference independent assumption **(A-6)** is accepted, it is fitting to assume that if $v_S(s)$ and $v_C(c)$ are determined for any landmark value of c and s, say c_L and s_L, respectively, shapes of these functions will be preserved throughout each region. $v_S(s)$ and $v_C(c)$ can be determined approximately by being given three points A, B, and C as shown on each function as shown in Fig.5.2. Whenever any policy alternative having purely positive or negative effect on their goal concept is adopted, value of c is identically equal to 1 for positive effect and -1 for negative one by Eq.(5-1). In this paper, we adopt $c_L = +1$ for the determination of $v_S(s)$ for $c > 0$ and $c_L = -1$ for $c < 0$. By doing this, the determination of $v_S(s)$ for $c > 0$ can be executed by asking the decisionmaker about the conditional evaluations on effect sensitivity, assuming a concerned policy concept or a set of policy concepts gives an purely positive effect on the goal concept, and $v_S(s)$ for $c < 0$ presuming a policy effects it in only suppressing ways. Let s_{max} be the calculated maximum impact for a certain selected goal concept. s_{max} is an impact in case all the policy concepts reachable to the goal concept are adopted as policy alternative[5]. As shown in Fig.5.2, the domain of $v_S(s)$ is a closed

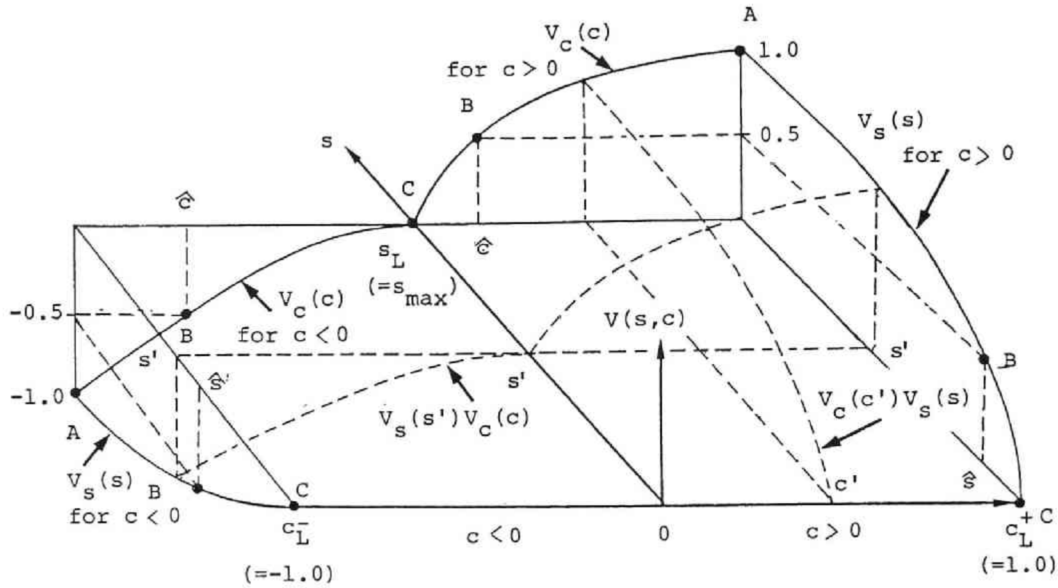


Fig. 5.2 Schematic representation of policy validity evaluation function $v(s, c)$.

interval $[0, s_{max}]$. Noting that impact s has the maximum value of s_{max} as mentioned above and takes the minimum value of 0 when no policy concept for the goal concept is adopted, we can assign the values of $v_S(s)$ for $c > 0$ and $c < 0$ at $s = 0$ and s_{max} as follows:

$$
\begin{aligned}
&v_S(s_{max}) = 1, \quad v_S(0) = 0 \quad \text{for } c > 0 \\
&v_S(s_{max}) = -1, \quad v_S(0) = 0 \quad \text{for } c < 0
\end{aligned}
\tag{5-11}
$$

They correspond to the extreme points, A and C of $v_S(s)$. The point B corresponding to $v_S(s) = 0.5$ for $c > 0$ is determined by asking the decisionmaker about a value of s, at which he thinks the difference of the

---

[5] $c_{max}=1$, $c_{min}=0$ and $s_{min}=0$ in any case, but s_{max} calculated according to the above mentioned assumption depends on the goal concept considered.

strength of preference from the upper extreme s_{max} case becomes equal to the one from the lower extreme 0, that is, about the decisionmaker's *equal difference point* $\hat{s}$ such that $(s = 0, c = 1)(\hat{s}, c = 1) \underset{\sim}{\ }^{*} (\hat{s}, c = 1)(s = s_{max}, c = 1)$, where the notation $wx \underset{\sim}{\ }^{*} yz$ means both $wx \overset{*}{\gtrsim} yz$ and $yz \overset{*}{\gtrsim} wx$. Point B corresponding to $v_S(s) = -0.5$ for $c < 0$ can be determined similarly.

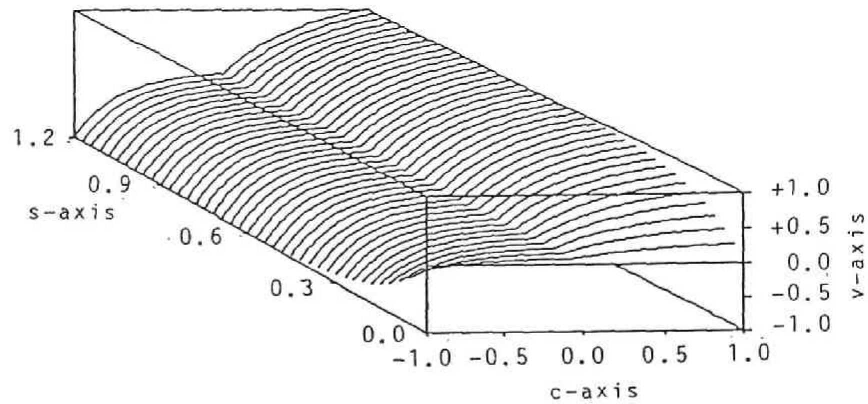
We adopt the above mentioned maximum impact s_{max} as s_L, the landmark value of s for the determination of $v_C(c)$. The domains of $v_C(c)$ are closed intervals $[0, c_L = +1]$ for $c > 0$ and $[0, c_L = -1]$ for $c < 0$, respectively. From the definition and consideration on a consonance measure c, we can assign the values of $v_C(c)$ at c = 0, 1, and -1 as follows:

$$v_C(c) = 1,\ v_C(0) = 0,\ v_C(-1) = -1$$

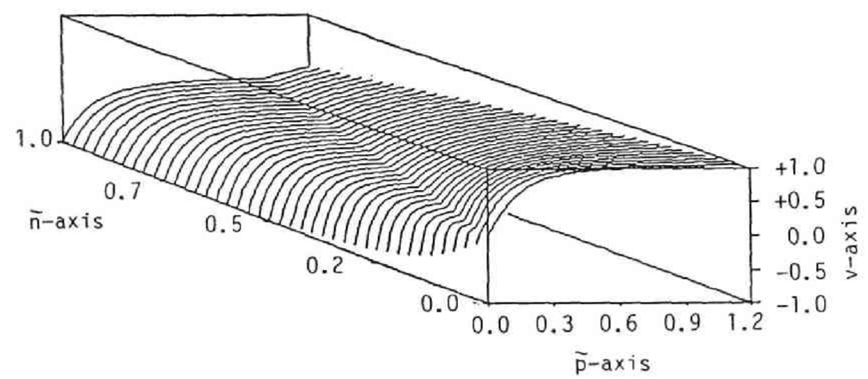
They correspond to the extreme points A and C of $v_C(c)$. Point B denoting the decisionmaker's equal difference point is determined in the same way with the determination of $v_S(s)$. The determination of $v_C(c)$ can be done by asking the decisionmaker about the conditional evaluation on the positive and negative composition of the effect when the effect is positive-predominant ($c > 0$) and the one when negative-predominant ($c < 0$), separately, assuming that effect has the maximum effect sensitivity s_{max} in both cases[6]. Fig.5.3(a) and (b) show policy validity functions represented three-dimensionally by the system over S x C and **P** x **N**, respectively.

---

[6] In this paper, as a continuous and monotonous function passing the three points A, B, and C, we used $v(x) = b(1 - \exp(-cx))$ and determined parameters b and c, so that the three points lie just on the function.



(a)



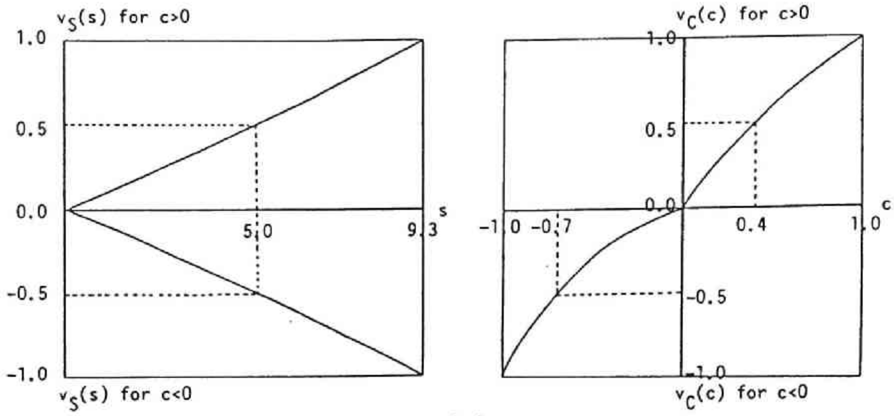
(b)

Fig. 5.3 Displayed policy validity evaluation functions over two kinds of attribute sets. (a) C x S space. (b) P x N space.

5.3.2 Identification of Policy Alternatives Based on Decisionmaker's Conformity

In the previous example "traffic problem in Japan," let us consider that three decisionmakers A, B, and C focused on improvement of "aggravation of traffic pollution (103)," that is, they selected (103) as system through question-answering processes between the decisionmakers and their common goal concept. There are nine concepts 6, 12, 47, 59, 72, 92, 99, 127, and 151, reachable to 103 as shown in the valency matrices of Fig.4.7(d). From the contents of those concepts, the eight concepts except 47 are considered as candidates for policy concepts which the decisionmakers can take for the goal concept (103). Fig.5.4, Fig.5.5, and Fig.5.6 show three types of $v_S(s)$ and $v_C(c)$ pair extracted from the decisionmakers having different preference characteristics from one another and corresponding rankings of the policy alternatives recommended by the the system. The preference characteristics of decisionmaker A can be interpreted as follows. His policy impact function $v_S(s)$ showing concave upwards for $c > 0$ and downwards for $c < 0$ implies that he expects the promoting impact of the policy alternatives to be conspicuously strong but he is not anxious about their suppressing impact unless it does not become conspicuously strong. His consonance function $v_C(c)$ showing convex upwards for $c > 0$ and $c < 0$ implies that he is satisfied with modest predominance of promoting impact which the alternatives have and he is not anxious about predominance of their suppressing impact unless it is not conspicuous.

The impact function $v_S(s)$'s extracted from decisionmakers B and C are convex upwards for $c > 0$ and downwards for $c < 0$ as shown in Fig.5.5(a) and Fig.5.6(a). This means that, in contrast to the attitude of decisionmaker A, both the decisionmakers do not expect so much promoting impact of the alternative but they are very anxious about its suppressing



(a)

VALUE	POLICIES 6	12	59	72	92	99	127	151
0.125	1	1	1	0	0	0	0	1
0.121	1	1	1	0	0	0	1	1
0.120	0	1	1	0	0	0	0	1
0.117	1	1	1	0	0	1	0	1
0.116	0	1	1	0	0	0	1	1

THE BEST POLICY SET IS AS FOLLOWS.

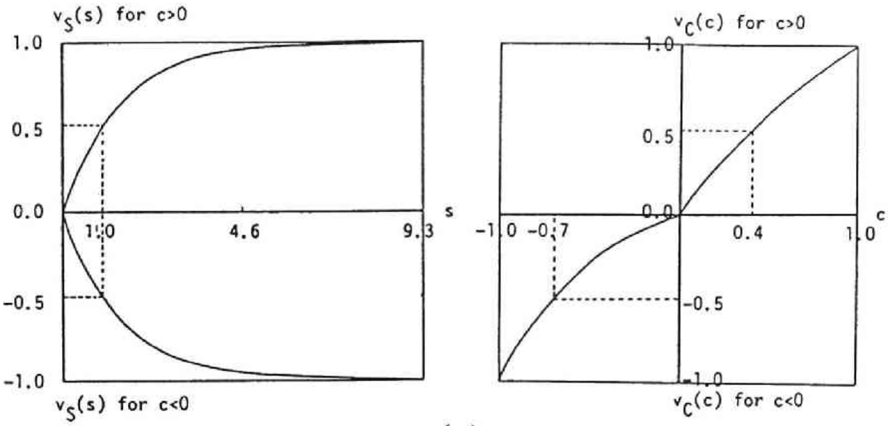
6.IMPROVE. OF TRAFFIC TERMINALS
12.SELF-IMPOSED CONTROL OF CAR USAGE
59.CONTROL OF PARKING CARS
151.SETTING NO TRUCK PASSING AREA IN DAYTIME

(b)

Fig. 5.4 Policy impact function $v_S(s)$ and policy consonance function $v_C(c)$, and recommended policy sets by the system for decisionmaker A.

impact. Decisionmaker B's consonance function $v_C(c)$ is similar to decisionmaker A's one, that is, implying modest preference for predominance of promoting power and not anxious about the containment of suppressing impact. Decisionmaker C's $v_C(c)$ having concave upwards for $c > 0$ and $c < 0$ implies that, in comparison with decisionmakers A and B, he expects the predominance of promoting impact to be conspicuous and he is very anxious about predominance of suppressing power. The three decisionmakers' preference characteristics can be summarized as shown in Table 5.1.

Fig.5.4(b), Fig.5.5(b), and Fig.5.6(b) show policies or sets of



(a)

VALUE	POLICIES 6	12	59	72	92	99	127	151
0.360	0	0	1	0	0	0	0	1
0.346	1	0	1	0	0	0	0	1
0.336	0	0	0	0	0	0	0	1
0.326	1	0	0	0	0	0	0	1
0.323	0	1	1	0	0	0	0	1

THE BEST POLICY SET IS AS FOLLOWS.

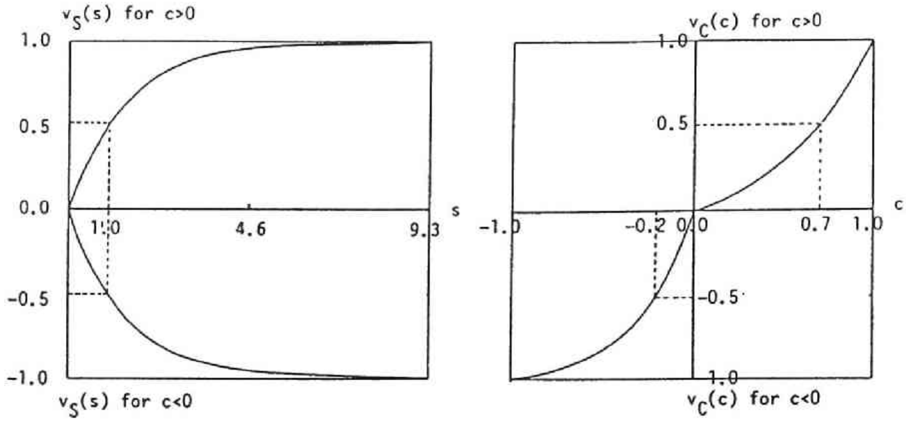
59.CONTROL OF PARKING CARS
151.SETTING NO TRUCK PASSING AREA IN DAYTIME

(b)

Fig. 5.5 Policy impact function $v_S(s)$ and policy consonance function $v_C(c)$, and recommended policy sets by the system for decisionmaker A.

policies, that is, policy alternatives, recommended by the system according to the preference characteristics of decisionmakers A, B, and C, respectively. In the figures, the first rows underlined represent the most approvable policy alternatives of the decisionmakers, and the second rows do the second best one, and so on. The left most columns indicate the values of the validity function v(s, c), and "1" and "0" means "adopted" and "not adopted," respectively. The most approvable policy alternatives of the decisionmakers A, B, and C are as follows.

Decisionmaker A: [6, 12, 59, 151], that is, simultaneous execution of



(a)

VALUE	POLICIES 6	12	59	72	92	99	127	151
0.141	0	0	0	0	0	0	0	1
0.138	0	0	1	0	0	0	0	1
0.128	1	0	1	0	0	0	0	1
0.127	1	0	0	0	0	0	0	1
0.113	0	1	1	0	0	0	0	1

THE BEST POLICY SET IS AS FOLLOWS.

151.SETTING NO TRUCK PASSING AREA IN DAYTIME

(b)

Fig. 5.6 Policy impact function $v_S(s)$ and policy consonance function $v_C(c)$, and recommended policy sets by the system for decisionmaker C.

"IMPROVE(MENT) OF TRAFFIC TERMINALS," "SELF-IMPOSED CONTROL OF CAR USAGE," "CONTROL OF PARKING CARS," and "SETTING NO TRACK PASSING AREA IN DAYTIME."

Decisionmaker B: [151, 59], that is, the policies "IMPROVE(MENT) OF TRAFFIC TERMINALS" and "SELF-IMPOSED CONTROL OF CAR USAGE" are excluded from A's policy set.

Decisionmaker C: [151], that is, only one policy "SETTING NO TRACK PASSING AREA IN DAYTIME" is recommended without others.

These sets are indicated by closed regions in Fig.5.1. Each decisionmaker's

	When decisionmaker's problem is improved ($c>0$)	When decisionmaker's problem is aggravated ($c<0$)
decisionmaker A	to prefer policy having reasonable impact, but not so sensitive to slight dissonance allowing some by-effects	not so harmed by policy having reasonable impact, and tolerant of policy whose resulted consequences are not clear
decisionmaker B	to prefer policy having even faint impact, and not so sensitive to slight dissonance allowing some by-effects	much harmed by policy having even faint impact, but tolerant of policy whose resulted consequences are not clear
decisionmaker C	to prefer policy having even faint impact, but much sensitive even to slight dissonance prohibiting any by-effects	much harmed by policy having even faint impact, and prefer avoiding any policy that may cause aggravation

Table 5.1 Summarized attitudes of three different decisionmakers A, B and C, for suppressing aggravation of traffic pollution (Concept 103)

preference characteristics indicated by Table 5.1 can be reflected well on the distribution of the regions in Fig.5.1 as well as in the result of recommended policies indicated in Fig.5.4(b), Fig.5.5(b), and Fig.5.6(b).

5.4 Concluding Remarks

As the recent societal and environmental problems get more highly complex, the necessity is increasing for the introduction of a decision support system which leads a decisionmaker to his final stage of decision process flexibly through the conversation with a knowledge base, i.e., a great amount of expertise specific to the problem domain. In societal complexes, a causal relationship such as the one between a policy and a goal, i.e., a sign of causal sequence between concepts, can be rarely determined as either positive or negative, but usually consists of both kinds of effects, that is, a policy promotes the attainment of a goal as well as suppresses it. This means that in judging the adoptation of policies, a decisionmaker should estimate the magnitudes of both kinds of effects individually and consider the adoptation as a trade-off problem between them.

In this chapter, based on the method for quantifying each kind of causal effects, we suggested a system whereby a decisionmaker can confirm his preference characteristics when he chooses a policy set among alternative policy sets affecting his own goals. This preference characteristics were discussed over two kinds of attributes: one was a policy impact representing a kind of controllability of a policy and the other one was a policy consonance, smallness of a kind of mental stress a policy with both favourable and unfavourable aspects charges the decisionmaker, in other words, "quantity" and "quality" of policy effects. Finally, we developed an methodology to extract the decisionmaker's trends of conformity above these two attributes and to derive the recommended alternative set of policy concepts accordingly.

PART III.

DSS BASED ON DEEP KNOWLEDGE STRUCTURES

In this part, to explicate the conceptual and sophisticated principles implicitly represented in the surface expressions, a methodology to organize hierarchical deep knowledge structures above the surface knowledge is developed. The knowledge organization through semantic meaning is much alike human memory structures and is closely related with their cognitive processes, especially with respect to concept formation from instances as well as to hypotheses-driven prediction. These are modelled based on the hierarchical causal knowledge structure and applied to an interactive DSS, which presents prediction of some events' occurrences or interpretation by adaptively reorganizing and generalizing instances of episodic knowledge through its deep knowledge structure. Such self-organizing capability is really valid in dealing with an open-ended problem domain and in making decisionmaker focus his attention in the uncertain, complicated reality.

CHAPTER 6

MULTI-LAYERED ORGANIZATION OF DEEP CAUSAL KNOWLEDGE

6.1 Introduction

As for a general architecture of DSS, a knowledge-based approach has been proposed, where the predecision steps should be supported, that is, helping the user identify and explicate the problem he is facing, through an interactive conversation which goes access to enormous amount of information. In this knowledge-based approach, DSS must have a store of domain-specific knowledge based on a collection of relevant historical cases or episodes consisting of a large variety of causally interrelated events, where various *actors*, that is, individuals, parties and organizations, are involved. Here, for the knowledge-based system, the key issue is not whether the sought information is eventually stored in the knowledge base or not, but how to support the users by presenting more conceptual and sophisticated principles implicitly represented in the superficially expressed information in the knowledge base. This means that DSS should have facilities for a filtering and a pattern recognition capability in order to search information selectively for the decisionmaker and to provide the user with a useful model, which is not a mathematically developed one but a roughly-simplified heuristical one. Such a conceptual model makes it possible to recognize problems more easily and to reduce the time necessary to give a preliminary analysis of the problem concerned.

For the construction of such a DSS, in this chapter, we at first develop a formalism to represent the empirical or episodic knowledge on socio-political problems written in documents as causally-chained networks. Our formalism is proposed so as to capture multiple perspectives including the behavioral, cognitive, and organizational aspects the socio-political problems inherently have. Above these networks, primitive syntactical units are defined as behavioral descriptors, which are necessary to identify the human being's universal sequences of behaviors, and are provided with conceptual meanings. Then, by defining concept formation rules as ordered sets of primitive units, a kind of meta-knowledge universally accepted as human social behaviors or the ways of interferences among actors, we aggregate a detailed causal network into a more essential sequence of larger conceptual units. Each of these larger units is an important repetitive planned subsequence of primitive units that entails themes, intentions, and outcomes of some social actions. Finally, from these simplified sequences, dynamically evolving patterns of mutual goal interactions among actors are detected, which is a sort of knowledge to which such a principle or a theory in cognitive social psychology has been addressed under a cognitive balance theory as a symbolic mental organization of aspects of social reality. By referring to its underlying knowledge structures in the hierarchy with different abstraction levels, it becomes clear how and when such a simplified principle comes into play (cf. Sawaragi, Iwai and Katai, 1985a, 1987b, 1987c).

6.2 Multi-Layered Organization of Knowledge for Deeper Understanding

6.2.1 Deep Knowledge for DSS

As Simon said, an ill-structured problem is defined as one putting demands on knowledge as past experiences, and what is considered as evidence of some judgment's truthfulness or plausibility in such a domain is the empirical knowledge itself rather than any deductive inferences seeking in an analytical way for some optimal solution (Simon, 1977). In such a domain, by accumulating his individual experiences or observations, the specialized decisionmaker can get an idea of what kinds of events naturally belong together or what sorts of general behavioral patterns appear in sequences of events, and these may suggest familiar courses of actions that are then edited or modified according to their rational investigation of consequences. The expertise, that such a specialized decisionmaker has, is actually a highly-developed capability of pattern recognition combined with a specific principle. This can be acquired through the accumulation of a large repertoire of context-specific patterns and by recognizing useful structural patterns within them. That is, those acquired behavioral patterns are by no means just a detailed traces of actual experiences or observations, but evolves in layers with a variety of abstraction levels. In this sense, we call these patterns "deep" knowledge (Chandrasekaran, 1984). In each layer, events develop in parallel concurrently under the domination of a social behavioral plan-scheme of the upper level. Human experts interpret new cases in terms of something with which they are already familiar, and when knowing some information does not fit into the available pre-existing pattern, they seek for such clues as similarities and analogy based on its structural pattern. In this way, they overcome sharp human cognitive limitations in the face of transient, unfamiliar, noisy and competitive information, and shape the reality of the world for their individual viewpoints.

6.2.2 Multi-Layered Organization of Knowledge

Indeed cognitive maps can convey a realistic model of the subtleties of thinking about a real complex world in a direct way and can be used for independent evaluations of whether some actor's policy is desirable or not to another actor, but what is represented in maps is just the structural appearance of the problem and has little bearing on the substantive problem nor is it any way to judge the difference or similarity among individual reasoning patterns. What we capture in our minds is by no means merely such a chain of individual social phemomena, but is a deeper conceptual knowledge than that.

Any social behaviour or action is done and developed under the domination of a more comprehensive social plan-scheme, where various kinds of interactions of themes and intentions of various actors come into play (Schank and Abelson, 1977; Wilensky, 1983) , although what is represented in causal structures looks like just linear temporal sequences of individual phenomena. Moreover, at the extreme level, i.e., the deepest level, there may exist a dynamic sequence consisting of such simple goal interactions as "conflict" and "agency," which forms a typical behavioral pattern of social phenomena (Abelson, 1975). The components of these patterns will capture substantive aspects at various abstraction levels, as a result of aggregating, or "rephrasing" the context of the lower components. The extraction and the presentation of such a typical behavioral pattern is what can be really called the "knowledge" that guides decisionmakers in a routine process for the problem solving. This knowledge organization makes it possible to explain the essential picture of how the world is going on, and contributes to removing the biases which produce a too restricted view and helps global understanding, all of which are fundamental facilities to be equipped for the realization of a man-machine system.

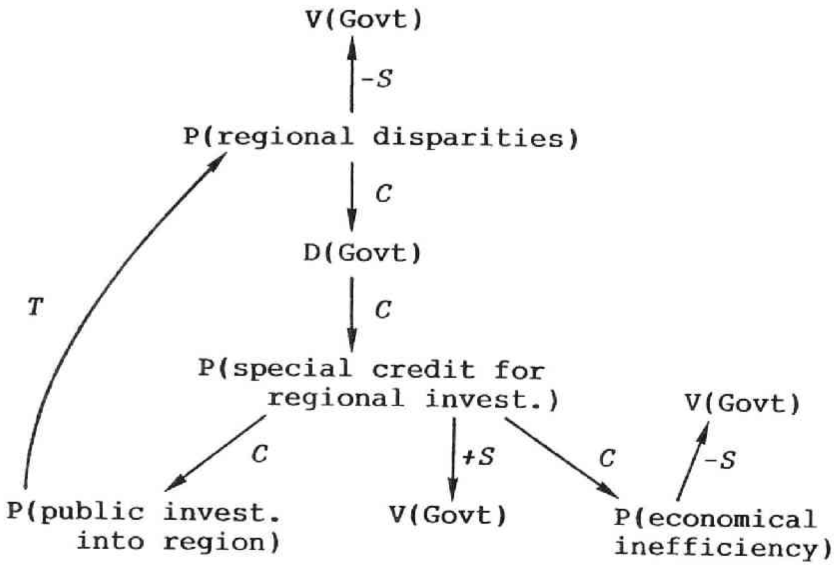
In order to organize systematically such a knowledge structure, it is required that a meta-knowledge of aggregation from the bottom to the top in the hierarchy as well as a formalism for representing causally-chained social events, which takes account of the minimum syntactics needed to identify the universal sequences of human behavior, are introduced.

6.2.3 Causal Network Model Representation

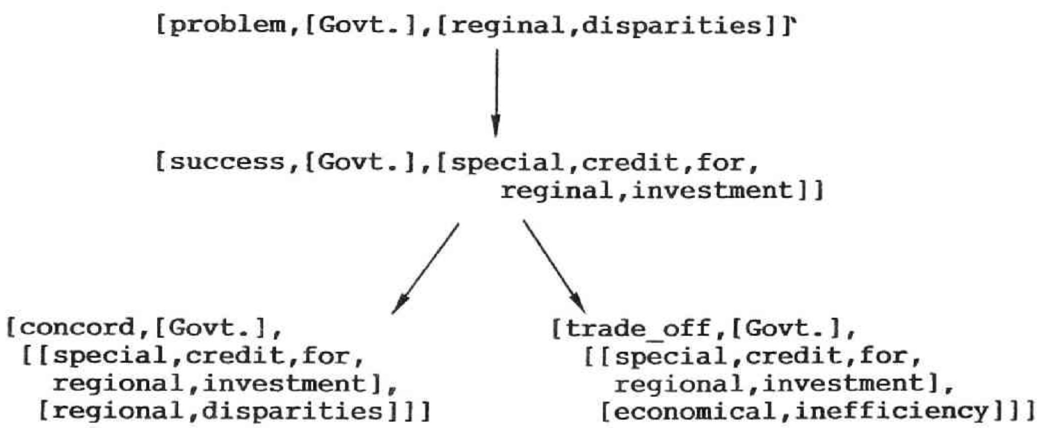
For the construction of such a multi-layered knowledge structure as mentioned in the previous section, the knowledge representation at the bottom level, that is, the representation formalism for encoding documented data, is very important. At the current stage, we are assuming that the construction of the knowledge base at the bottom level is done by our manual coding, while the other processes are performed by the system automatically. Therefore, what is required in the process of encoding are: first, it must be both easy and general in the sense that anyone without any specialized skills can encode the document data, and the encoded result should not generate any significant differences among the coders who encode the same document data. Second, and this is more important for the theme in this paper, the encoded knowledge must be so well-formalized that the system can process the aggregation of the encoded knowledge towards upper patterns both smoothly and efficiently capturing such deep meanings as described in the preceding chapter.

Cognitive maps satisfy the first criterion, since Axelrod and others have established the detailed instructions and specifications for encoding the maps from documentary statements (e.g., Axelrod, 1976) and our previous paper verifies the objectivity or consistency of the encoded results among different actors (Nakamura, Iwai and Sawaragi, 1982b). As for the second criterion, however, in so far as we extract concepts that are explicitly expressed in the documents according to the cognitive mapping, we may miss capturing such essential components as the intervention of actors' intentions or decisions and the dynamic interactions among actors' interests or beliefs. Indeed, in the documented statements, those features are not always expressed so explicitly that they can be extracted as concrete descriptions like *concepts* of cognitive maps, but can be recognized easily in the context, and these must be made explicit in the coding process for the construction of the deep knowledge structures.

The alternative knowledge representation formalism (called *Causal Network Model Representation*, *CNM* abbreviated), which we propose in this thesis, is causally-chained network representation consisting of nodes and linkages as in the conventional cognitive maps. Different from the conventional cognitive maps, we classify nodes into the following three types: *Decision Nodes (D nodes)*, Perception Nodes (P nodes), and *Value or Belief Nodes (V nodes)*. Moreover, as for linkages between those nodes, we define two kinds of causal linkages, *Cause relations (C relations*, abbreviated) and *Terminate relations (T relations)* as well as two sorts of *Sentiment relations*, *positive sentiment relations (+S relations)* and *negative ones (-S relations)*. For instance, the following description of documents:



(a)



(b)

Fig. 6.1 Illustration of causal network model representation and extracted primitive events

"Taking the regional disparities of economy serious, Government decided to offer the special credit for regional investment. As a result, public investment into regions increased and the situation was improved, but at the same time brought about another problem of economical inefficiency for Government."

can be represented with those kinds of nodes and linkages as a network as shown in Fig.6.1(a), whereas,

(N-1) Perception Nodes (P): these represent concrete contents of behaviors or events, that is, kinds of world aspects or states subject to change. The node P("regional disparities of economy") in Fig.6.1(a) illustrates this type of node.

(N-2) Decision Nodes (D): this type of nodes is the one for explicating the times of interventions of some actor's decisions or intentions. As D(Govt) in Fig.6.1(a) illustrates, D node is a kind of dummy node, which is encoded from the documents with names of actors without any concrete contents (concrete contents of those decisions are encoded as P nodes).

(N-3) Value or Belief Nodes (V): these represent actors' abstract beliefs or interests and are also encoded with the names of actors as D nodes. For instance, in Fig.6.1(a), the node V(Govt) connected with the node P("regional disparities of economy") represents actor Government's values, which is stated in the statement "taking the regional disparities of economy serious, ...".

As for connected linkages between D and P, or P and P, the following two kinds of causal relationships are defined;

(L-1) Cause Relations (C): the content of the preceding node causes or initiates one of the following nodes.

(L-2) Terminating Relations (T): the content of the preceding node suppresses or disables the one of the following nodes.

Some of the P nodes are directly connected with V nodes, and the relations between them represent sentiment relations rather than carrying the causality as C and T relations.

(L-3) Sentiment Relations (+S and -S): the positive sentiment relation (+S) represents that the content of the preceding P node contributes to the values of the actor of the following V node, and the negative one (-S) does vice versa.

Causality is propagated through C and T relations via P and D nodes but not V nodes, since +S and -S relations connected to V nodes carry only sentiment relationships, and no linkages go out of V nodes.

6.3 An Aggregation of Causal Network Model Representation into Semantically Integrated Events

6.3.1 Aggregation of Causal Network Representation into Primitive Events

The configurations of possible connections among nodes determined as syntactically rational ones from an intuitive perspective are summarized as shown in Table 6.1. These typologies of nodes and linkages make it possible to recognize any metaphorically similar or equivalent causal structures of events, on the basis of their determining structural or syntactical equities apart from their entailing semantic contents. That is, some ordered sets of categorized linkages and nodes can imply the structural meaning that is universally true whatever contents or actors those component nodes may have. This is because our recognizing social events from human societal behavioral perspectives can be universally defined by the combinations of those purely syntactical elements such as decisive actions, resulting changes in social aspects, and emotional

consequent node / antecedent node	D	P	V
D	$C \cdot T$	C	
P	C	$C \cdot T$	$+S \cdot -S$
V			

Table 6.1 Configuration of possible linkages among nodes

responses both positive and negative.

By the way, the reason why behavioral patterns with various degrees of complexity can be recognized with reasonable conceptual meaning from the sequence of categorized nodes, is that we originally have a small number of normalized events as conceptual units. Those units are regarded as *primitives* in the sense that they cannot be further decomposed into smaller significant units, and our recognizing societal events are to be defined as a specific set of those primitives.

In general, there exists no rational principle in defining such a set of primitives, but from the viewpoint of the performance of the system, it is required (i) that they be kept to as a small number as possible and able to be recognized easily, and (ii) that they are capable of defining smoothly more complex event concepts by their combinations.

The research on these *semantic primitives* has been developed especially in the field of artificial intelligence with respect to the natural language processing as well as to the planning problems. For instance, Schank and others introduced a set of *primitive ACTs* and a set of *BASIC SOSIAL ACTs* for describing physical and social behavior, respectively (Schank, 1972; Schank and Carbonell, 1978). Abelson developed a set of *DELTACTs* for understanding plans implied in social behavior (Abelson, 1975). Lehnert defined a set of *PLOT UNITs*, which were determined based on the emotional responses to others' behavior and used for understandig tales or stories (Lehnert, 1981, 1982).

Here, based on the above-mentioned criteria, we consider the nine cause-effect relationships as shown in Table 6.2(a) between two out of three kinds of nodes, that is, D nodes and two kinds of P nodes connected to V nodes through +S and -S relations satisfying the configurations of Table 6.1, which work as unitary structures offering the above mentioned primitive meanings. According to whether the two actors concerned with an antecedent cause node and a consequent effect node are identical or not, each cause-effect relationship provides a different conceptual meaning. We define those as *intra-actor* primitive events when identical and *inter-actor* ones when different, as shown in Table 6.2(a) and (b), respectively.

The elements listed in the upper six rows in Table 6.2 are the ones where a decision has intervened in some form, that is, whose unitary structures involve D nodes.

(1) "**problem**": an event where the occurrence of an aspect X undesirable to actor A (Node P(X) connected to node V(A) through -S relation) causes him to decide (node D(A) preceeded by P(X) through C relation), that is, an event representing "the consciousness of a problematic situation."

	(a) A = B	(b) A ≠ B
$V(A) \xleftarrow{-S} P(X) \xrightarrow{C} D(B)$	[problem, [A], [X]]	[inducement, [A, B], [X]]
$V(A) \xleftarrow{+S} P(X) \xrightarrow{C} D(B)$	[enablement, [A], [X]]	[promotion, [A, B], [X]]
$D(A) \xrightarrow{C} P(X) \xrightarrow{+S} V(B)$	[success, [A], [X]]	[commitment, [A, B], [X]]
$D(A) \xrightarrow{C} P(X) \xrightarrow{-S} V(B)$	[failure, [A], [X]]	[threat, [A, B], [X]]
$D(A) \xrightarrow{C} D(B)$	[motivation, [A], []]	[mental_trans, [A, B], []]
$D(A) \xrightarrow{T} D(B)$	[abandon, [A], []]	
$V(A) \xleftarrow{a} P(X) \overset{b}{\dashrightarrow} P(Y) \xrightarrow{c} V(B)$ $a*b*c = -1$	[trade_off, [A], [X, Y]]	[competition, [A, B], [X, Y]]
$V(A) \xleftarrow{-S} P(X) \overset{b}{\dashrightarrow} P(Y) \xrightarrow{c} V(B)$ $b*c = -1$	[dilemma, [A], [X, Y]]	[adversity, [A, B], [X, Y]]
$V(A) \xleftarrow{+S} P(X) \overset{b}{\dashrightarrow} P(Y) \xrightarrow{c} V(B)$ $b*c = +1$	[concord, [A], [X, Y]]	[cooperation, [A, B], [X, Y]]

Table 6.2 Primitive events. (a) Intra-actor ones. (b) Inter-actor ones.

(2) "**enablement**": contrary to "problem", an event where an existence of an aspect X desirable to actor A (node P(X) connected to node V(A) through +S relation) prompts him to decide (node D(A) preceded by P(X) through C relation).

In these unitary structures, when the actor of the subsequent D node is replaced by a different actor from the one of its antecedent node, these two primitive events turn out to be "inducement" and "promotion", respectively, both of which represent events where the situation faced by one actor causes another actor to decide something.

(3) "**success**": an event where as a result of actor A's decision (D(A)) something in line with his belief or will follows (node P(X) is connected to node V(A) through +S and preceded by D(A) through C relation).

(4) "**failure**": contrary to the event "success", something against actor A's belief or interest follows.

For these two elements, the corresponding inter-actor primitive events are "commitment" and "threat", respectively, representing events of some actor's decision with consideration on the contribution or discontribution to another actor's interests.

(5) "**motivation**": an event where actor A's decision (consequent node D(A)) is motivated as a result of one of his previous decisions (antecedent node D(A) and its outgoing C relation).

(6) "**abandon**": an event where a decision previously made by actor A (consequent node D(A)) is canceled (T relation between D(A)'s) as a result of his another of his decisions (antecedent node D(A)).

The inter-actor primitive events with regard to "**motivation**" are "**mental_**

trans," which represents an event of two different actors' mental contact, that is, an event where the antecedent actor's will is transfered to the subsequent actor.

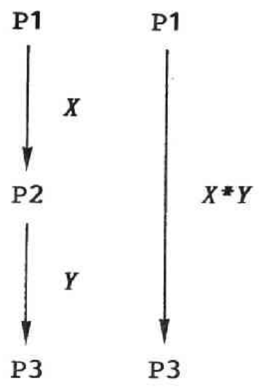
The unitary structures in the lower three rows of Table 6.2 consist of chained non-intentional aspects of P nodes, where the initiating and the ending P nodes are connected to V nodes , representing goal interaction relationships within one actor or between different actors.

(7) "**trade-off**": a situational event where both desirable and undesirable aspects coexist, causally connected with each other. This event of "trade-off" covers four kinds of cases, that is, a desirable aspect causes an undesirable one ($V(A) \xleftarrow{+S} P(X) \xrightarrow{+1} P(Y) \xrightarrow{-S} V(A)$) or terminates a preexisting desirable one ($V(A) \xleftarrow{+S} P(X) \xrightarrow{-1} P(Y) \xrightarrow{+S} V(A)$), or an undesirable aspect causes a desirable one ($V(A) \xleftarrow{-S} P(X) \xrightarrow{+1} P(Y) \xrightarrow{+S} V(A)$) or terminates a preexisting undesirable one ($V(A) \xleftarrow{-S} P(X) \xrightarrow{-1} P(Y) \xrightarrow{-S} V(A)$), all of which are represented compactly using the multiplication operation "*" in Table 6.2. In this multiplication operation, the indirect relation through a number of P nodes as shown in Fig.6.2 can be determined by the ordinary multiplication, where the linkages C and T are regarded as +1 and -1, respectively.

(8) "**concord**": an event of the occurrence of desirable aspects in succession.

(9) "**dilemma**": an event of the occurrence of undesirable aspects in succession.

These above events denote interactions within one actor, but when such interactions happen between different actors, they are refered to as "competition," "cooperation" and "exclusion," respectively, and they work



(a)

*	C(+1)	T(-1)
C(+1)	C(+1)	T(-1)
T(-1)	T(-1)	C(+1)

(b)

Fig. 6.2 Determination of indirect causal effect

as important clues for detecting more global evolutional behavioral patterns in which many actors participate as mentioned in the following sections.

Fig.6.1(b) shows illustrations of primitive events, "problem," "success," "concord," and "competition," all of which are found in the causal network model representation of Fig.6.1(a). In this figure, each primitive event is found specified with its list of actors contents and component nodes, all of which are filled with the encoded node descriptions of the causal network representation, and they are represented as predicates having those specifications as their arguments. The actors list consists of one actor for the intra-actor primitive events and of two actors for the inter-actor ones, while the contents list is filled with dual contents for the primitive events concerned with goal interactions, that is, the ones in the lower three rows in Table 6.2, and a unique content otherwise [1]. The correspondences between the arguments of the predicates and the node descriptions of their causal network representation are defined rigidly as shown in Table 6.2, where the capital letters in each predicate represent the entities of its unitary structure [2].

6.3.2 Aggregation of Sequences of Primitive Events into Complex Events

As shown in Fig.6.1, it now becomes possible to rewrite what is represented by three kinds of nodes and four sorts of relations of causal network model into an evolutional sequence of primitive events. However, it is still too micro as compared with what we understand intuitively and conceptually in reading each statement corresponding to the causal network model representation of Fig.6.1(a). It is natural that frequently-appearing ordered sequences of primitive events are considered as being integrated as a whole into larger conceptual chunks. Actually in the domain of psychology, it is widely accepted that what restricts information capacity for the human short term memory is the number of chunks as semantic units rather than absolute information bits in a mathematical sense (Miller, 1956).

In our dealing socio-political domain, where various kinds of human factors are concerned, those chunks are considered to be predetermined routinized sequences of steps to be taken that involve not only an actor's eventual decisions but also his purposes or intentions, and the

---

[1] The descriptions of intermediary P nodes are omitted from the specifications of primitive events, as they are considered non-essential.

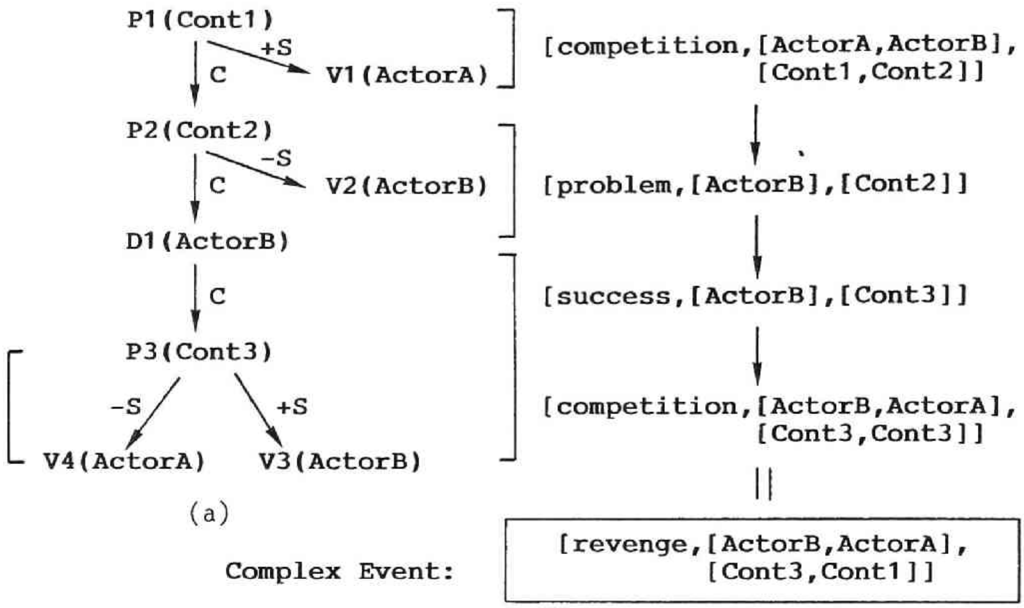
[2] The definition may accord with the structure of $V(A) \leftarrow P(X) \rightarrow V(B)$ where P1 and P2 are identical in their unitary structures. In this case, the arguments of dual contents are filled with the identical contents. The primitive events of cause-effect relationship between D nodes, that is, "mental-trans," "abandon," and "motivation" cannot have any contents since their composing D nodes are dummy ones without concrete contents. So the contents list for those is nil [].

results produced or the other actor's resulting responses to those decisions. For instance in Fig.6.1(b), a subsequence of primitive events "problem" "success" "concord" is extracted, which is a typical sequence of events to occur when an actor intentionally resolves the problem, and it can be aggregated into the more comprehensive chunk of "intentional-resolution" as a whole. In contrast to the primitive events, we call such a larger conceptual unit a *complex event*. What is important here is that such a complex event can be defined syntactically as the combination of primitive events. Therefore, whatever actors or contents its component primitive events may have, the complex events hold true universally as far as they satisfy constraints on their causal ordering and the consistency among entities is maintained.

In this section, we define a number of complex events as ordered sets of primitive events[3]. These complex events are classified into *inter-actor* complex events and *intra-actor* ones depending on whether or not they involve inter-actor primitive events as part of it or not.

These complex events are classified into inter-actor complex events and intra-actor ones depending on whether they involve inter-actor primitive events as part of it or not. Fig.6.3 illustrates a definition of an inter-actor complex event, "revenge." The causal network representation of Fig.6.3(a) and the extracted sequence of primitive events from it in (b) represent the following flow of events: the conflictual situation between actors A and B concerning issues of P1 and P2 (competition([A,B],[P1,P2],

[3] The number of complex events cannot be determined definitely, but is to be defined according to the purpose of the system in use in its domain. In our case, for decision support in the socio-political domain, several events necessary for the structural analysis of the problem are defined mainly concerning trade and communication activities observed repeatedly in our experiences.



(a)

(b)

```
revenge([ActorB,ActorA],[Cont3,Cont1]) :-

        competition([ActorA,ActorB],[Cont1,Cont2]),
        problem([ActorB],[Cont2]),
        success([ActorB],[Cont3]),
        competition([ActorB,ActorA],[Cont3,Cont3]).
```

(c)

Fig. 6.3 Definition of complex events "revenge"

[[P1,V1],..,[P2,V2]])) causes the problematic situation for actor B on P2 (problem([B],[P2],[[P2,V2],[D1]])), which makes actor B decide P3 in line with his belief (success([B],[P3],[[D1],[P3,V3]])), and as a result of that, actor A suffers from P4 caused by actor B's performing P3 (competition([B,A],[P3,P3],[[P3,V3],..,[P3,V4]])). This sequence of primitive events describes typical social behavior of revenge or retaliation occurring between two actors, and can be aggregated into a larger conceptual unit as a whole meaning "actorB's revenge on actor A for A's P1 by doing P3 (revenge([B,A],[P1,P3],[[P1,V1],..,[P2,V2],[D1],[P3, V3],..,[P3,V4]])). Therefore, the definition of the complex event "revenge" can be written as an ordered set of primitive events as shown in Fig.6.3(c), which is represented as a horn-clause of PROLOG[4]. Here, capital letters represent variables that can match any entities and the same letters bind the same entities. This binding represents the constraints on causal connections among primitive events or appropriate correspondences among entities of the complex event and its component primitive events. Table 6.3 shows the list of complex events in a form of horn clauses prepared in the current version of the system so far.

The complex events do not always consist only of primitive events. For instance, consider the causal network representation shown in Fig.6.4(a), which is similar to the one of Fig.6.3(a), but differs from it in the enclosed partial causal structure. That is, it describes that suffering from the conflictual situation with actor A, actor B gets mental

[4] In DEC-10 type PROLOG, a term preceding the notation ":-" is called a "head" part and a set of terms following it is a "body" part. The predicate in a "head" part represents a goal to be verified and the following set of predicates in conjunction in a body part denotes subgoals for the verification of the goal. In Fig.6.3(c), a complex event is located in a head part and its component primitive events are in a body part.

(a) intra-actor		resolution :- problem → success* → concord goal_pursuit :- enablement → success* → concord recovery :- failure → problem → success* ....
(b) inter-actor	con-sistent	compensation :- *competition* → enablement → *inducement* → *competition* → *cooperation* relief :- inducement → success* → *cooperation* accepted_request :- *mental_trans* → success* → *competition* ....
	incon-sistent	revenge :- *competition* → problem → success* → *competition* obstruction :- *promotion* → success* → *competition* rejected_request :- *mental_trans* → success* → *competition* successful_enforcement :- *threat* → problem → failure* → *competition* ....
	mutually contra-dictory	failed_request :- *mental_trans* → failure* → *adversity* ....

Table 6.3 Some of complex events. (a) Intra-actor ones. (b) Inter-actor ones.

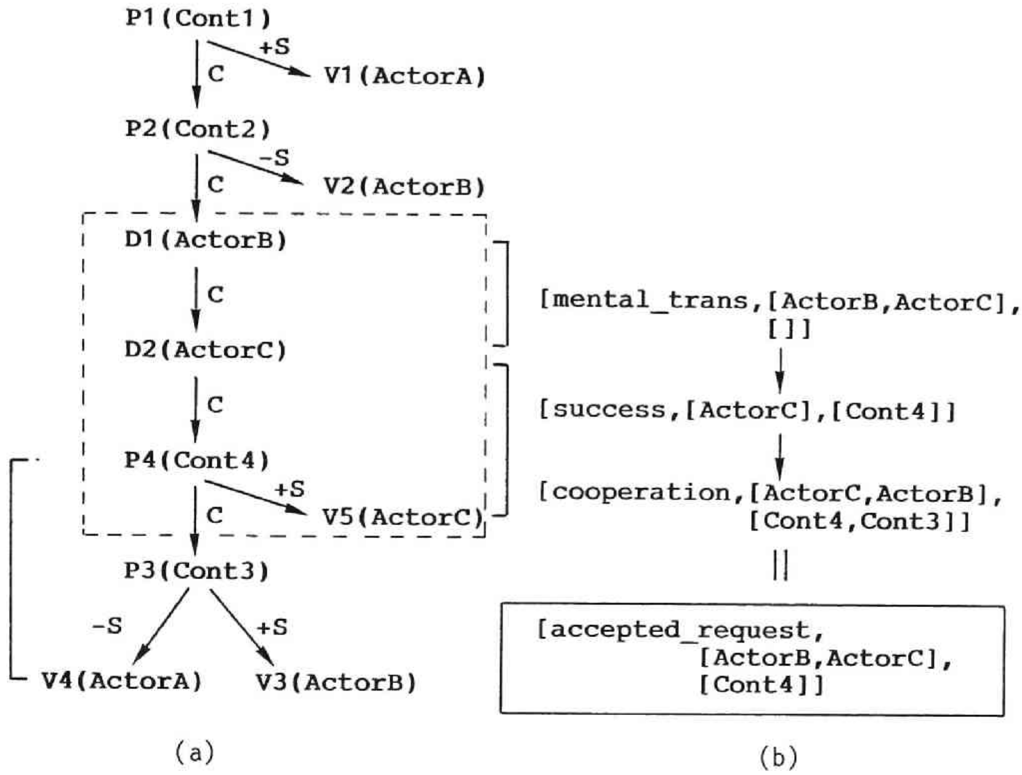


Fig. 6.4 Illustration of complex event functioning as instrumental means

contact with another actor C, and C's resulting decision is welcomed by actor B, which produces a conflictual situation between actors B and A. From this enclosed part, a sequence of primitive events, "mental-trans" "success" "cooperation" is extracted, which forms another complex event of "accepted-request" according to the definition of Table 6.3. This complex event plays an alternate or instrumental role in actor B's event of "success" in Fig.6.3(b) in its larger context of complex event "revenge." That is, the complex event of actor B's "revenge" on actor A is performed indirectly by replacing its component primitive event "success," which was originally to be attained by actor B himself, by another complex event including actor C. Such a component primitive event replaceable with another complex event is either "success" or "failure," and it is denoted by superscript "*" in the definition of Table 6.3 [5].

6.4 Evolving Patterns Implied in Causal Network Model Representation

6.4.1 Extracting Sequences of Complex Events from Causal Networks

The recognition of complex events in CNM is done in a top-down way, that is, for all complex events it is tried one by one whether its component primitive events are found in a given CNM satisfying the constraints for the variable bindings. Compared with a bottom-up searching method, where at first all possible primitive events are found and then an attempt is made to see whether some of these form complex events or not, such a top-down approach is more efficient, since from the encoded CNM so many primitive events are found that it is inefficient to try all their possible combinations.

Fig.6.5(a) shows a network representing the following description of some expert's view on the problem of Japan-US trade imbalance by CNM:
(JSI: Japan Steel Industries, USS: US Steel Co., LSC: League of Steel Congressmen)

[5] Note that it is not true that any complex events can replace "success" or "failure." The complex events to be replaced with "success" are restricted to the consistent complex events, and the ones for "failure" are either the inconsistent ones or the mutually contradictory ones. This will be mentioned in more detail in the subsequent section.

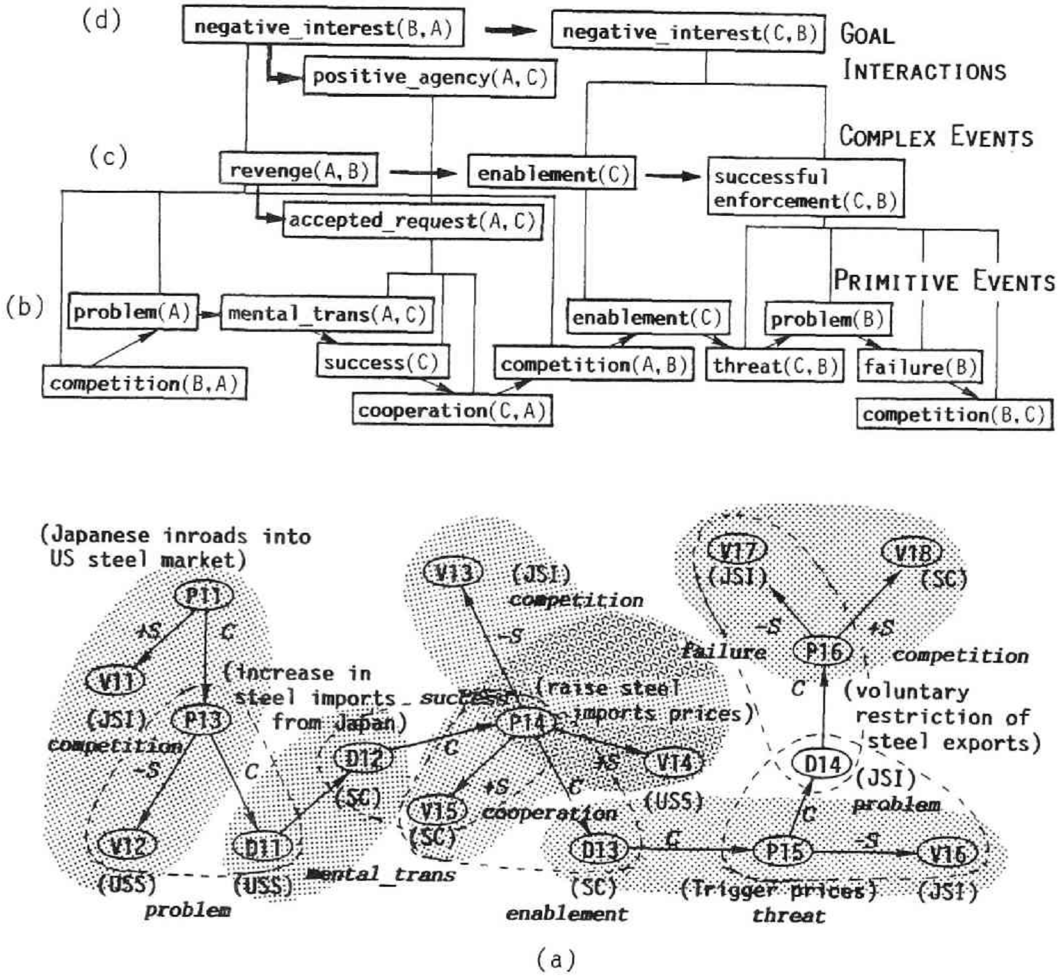


Fig. 6.5 Illustration of organized multi-layered knowledge structure. (a) Causal network model. (b) Primitive events. (c) Complex events. (d) Goal interaction relationships.

"JSI's inroads into US steel market (P11->V11) increased the steel exports of Japan to US (P11->P13), which did harm against USS (P13->V12). Then, USS petitioned to LSC and LSC supported USS's request of raising steel imports prices (P13->D11->D12->P14->V15), which was welcome by USS (P14->V14) and forced a significant burden on JSI (P14->V13). In reply to this, LSC decided to set the Trigger Prices, which obliged JSI to abide by that and had to perform the voluntary restriction of steel exports against its will (P14->D13->P15->D14 ->P16->V17)."

For this causal network, the complex event formation rules as an ordered set of primitive events shown in Table 6.3 is applied. That is, the rules are applied one by one whether or not all of their component primitive events are found on the CNM. For instance in Fig.6.5(c), the complex event "accepted_request" is found, since the causal structure in the matter includes the ordered sets of primitive events; US Steel Co.'s "mental_trans" to League of Steel Congressmen (D11(A) -> D12(C); A:US Steel Co., B:League of Steel Congressmen), "success" of LSC on acceptance of raising steel prices (D12(C) -> P14(X1) -> V15(C); X1:acceptance of raising steel prices) -> V15(C)), and a state of "cooperation" between LSC and US Steel Co. on this matter (P14(X1) -> V14(A), P14(X1) -> V15(B)), all of which satisfy the complex event formation rule for "accepted_request" in Table 6.3. This complex event of "accepted_request" can be regarded as an instrumental means of actor A's global planned behaviour of "revenge" on another actor C. Namely, a state of "competition" between actor A and B on the issues of Japanese inroads into the US steel market (X2) and the increase in steel imports from Japan (X3) (P11(X2) -> V11(B), P13(X3) -> V12(C)) causes the event of "problem" to actor A. Then, instead of actor

A's executing "success," the event of his "accepted_ request" to actor C occurs, and as a result of that, a state of "competition" between actor A and C on the issue of acceptance of raising steel prices (X1) (P14(X1) -> V14(A), P14(X1) -> V16(C)) is caused. That is, the rule for "revenge" is satisfied by replacing its component "success*" with "accepted_request."

As this example shows, the searching of the complex events is done in a top-down way rather than in a bottom-up fashion, in which all possible events existing in the CNM representation are extracted and then checked to find whether some of these form the complex events. Indeed, it may happen that some primitive events are missed in the searching bacause they are not aggregated into any larger complex events such as the one of "competition" between actor B and C on the issue of acceptance of raising steel prices (P14(X1) -> V14(B), P14(X1) -> V15(C)) in the figure. However, as far as the global view governing the whole causally-chained network is concerned, it may be neglected as a local one in the sense that it does not play a role in forming any complex events. Therefore, the top-down approach contributes to explaining the lower level event essential to its dominating evolutional flow of events at the upper level, as well as to avoid the combinatorial explosion in finding the complex events caused by extracting even such local primitive events.

At the complex event levels of Fig.6.5(c), another event of c_ev(14): "enablement" is also extracted. Although "enablement" is actually a primitive event which is not a part of any complex event, it relays between the already extracted complex events, in this case, c_ev(13): "revenge" and c_ev(11):"successful-enforcement." So it plays an essential role in maintaining causal connection of overall event flow, we deal with such a primitive event as a complex one exceptionally[6].

c_ev(11): accepted_request
c_ev(12): revenge
c_ev(13): successful_enforcement
c_ev(14): enablement

	c_ev(11)	c_ev(12)	c_ev(13)	c_ev(14)
c_ev(11)	0	0	0	1
c_ev(12)	2	0	0	1
c_ev(13)	0	0	0	0
c_ev(14)	0	0	1	0

(a)

scenario(1):

c_ev(12) → c_ev(14) → c_ev(13)
c_ev(12) → c_ev(11)

scenario(2):

c_ev(11) → c_ev(14) → c_ev(13)

(b)

Fig. 6.6 Illustration of adjacent matrix and extracted episodic scenarios.

These extracted complex events can constitute a causal temporal sequence, the ordering of which is determined by referring to the overlap among their component nodes on the CNM representation. Fig.6.6(a) shows a

[6] The primitive event that can be dealt with as a complex event in the above described situation is restricted to intra-actor primitive events.

determined adjacent matrix $[a_{ij}]$ among the complex events in Fig.6.5(c), where $a_{ij} = 1$ when complex event c_ev(i) directly precedes c_ev(j) causal-temporally, and $a_{ij} = 2$ when c_ev(j) functions as an instrumental means of c_ev(i). The complex events c_ev(j)'s such that $a_{ij} \neq 1$ for all i denote the initiating complex event of the sequence, and there exist as many sequences as those initiating complex events. We call this sequence of complex events determined from the adjancent matrix an *episodic scenario*. Fig.6.6(b) illustrates two episodic scenarios extracted from the matrix of (a).

6.4.2 Extraction of Goal Interaction Relationships Implied in Sequences of Complex Events

The episodic scenarios reveal evolutional flows of events capturing essential aspects of the apparent complex world reality. Unlike the detailed network representation of CNM, since these scenarios are modified with kinds of event concepts they are easy both to grasp in brief and to understand intuitively, although the former formalism is rather convenient in the process of encoding knowledge from the documents. With more actors involved in the problem and the more individual actor's policies or goals are interwoven with each other, the more the problem becomes complex. So it seems plausible to consider another level of representation. Actually in facing such a problem and judging whether some policy should be adopted or not, a decisionmaker at first attempts to predict its consequences especially on whom that policy does a favour and who are damaged by that policy, which suggests that the explication of ways how individuals intervent with other's goal, that is, *goal interaction relationships*, are of great effect for supporting decisionmakers in envisioning their political scenarios.

In principle, such goal interactions occur when two different actors participate in the same complex events, and evolve according to the alterations of actors along a sequence of complex events, i.e., episodic scenarios.

In the scenario(1) of Fig.6.6(b), the initiating event c_ev(12): "revenge" implies a state caused by the conflicting goals of two actors, "US Steel Co." and "Japan Steel Ind" on the trade situation. This goal interaction relationship turns to another conflictual state between actors "Japan Steel Ind" and "LSC," which is implied in the complex event of c_ev(13):"successful_enforcement" following the "revenge" through a unique actor "LSC"'s "enablement" on "setting Trigger prices." Contrary to those, the complex event c_ev(11): "accepted-request" implies that actor A asks actor B for the attainment of his goal and thus is successfully performed, that is, there is a state of interaction of cooperative goals between them. Since in this case the complex event of "accepted-request" is performed as an instrumental means of some other complex events, we regard such an interaction as successful cooperation between them.

The extraction of goal interaction relationships from scenarios is executed as follows: noticing at first any inter-actor complex event between actor A and B in the scenario it is then checked to see whether in a directly preceding or subsequent position in the script there exist complex events with a unique actor, either actor A or B, that is, intra-actor complex events or ones formed with a unique intra-actor primitive event such as an "enablement" of scenario(1). If these exist, we regard a goal interaction relationship to be occurring in the sub-sequence of complex events whose actor is either A or B. This is illustrated in Fig.6.7(a) and (b). As for the types of goal interactions, the following

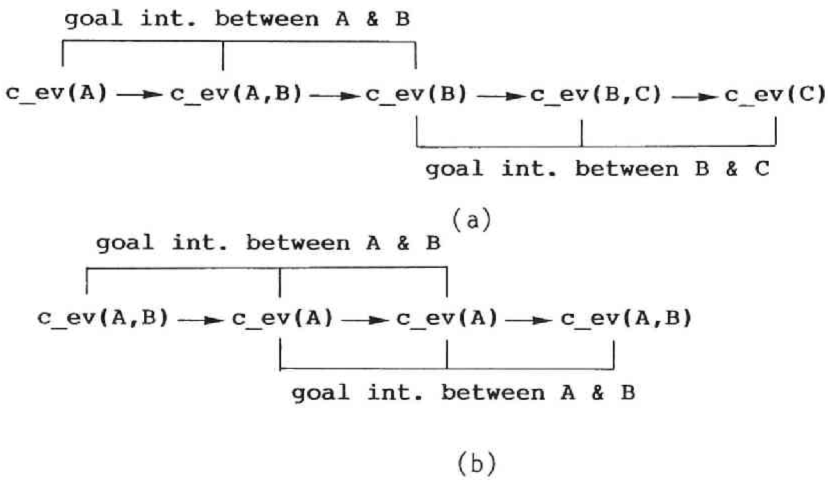


Fig. 6.7 Extracting goal interaction relationships from episodic scenarios

three "interest-relations" are defined: a state of cooperation "positive-interest" ("pos-int," in brief), a state of conflict "negative-interest" ("neg-int"), and a state of antagonism "antagonistic-interest" ("ant-int"), which are determined according to the classification of the inter-actor complex event; "consistent," "inconsistent," and "mutually-homicide" relations, as shown in Table 6.4. As for an inter-actor complex event functioning as an instrument of another complex event, there exists no adjacent event, and we consider interactions occurring there as "agent-relations" representing that an actor of the latter complex event depends on the execution of the attainment of his goal on another actor's performance. These relations defined for such complex events are categorized into two classes "positive-agent" ("pos-agt," in brief) meaning successful agent relations and "negative-agent" ("neg-agt") denoting failed

Complex events	Interest	Agent
Consistency	positive_interest	positive_agent
Contradiction	negative_interest	negative_agent
Mutual-Homicide	ant_interest	

Table 6.4 Classifications of goal interaction relationships implied by inter-actor complex events

ones, according to whether the complex event is replaced with "success", i.e., "consistent" complex events, or with "failure", i.e., "inconsistent" or "mutually-homicide" ones in Table 6.4.

6.5 Concluding Remarks

Human experts' decisionmaking is characterized by their simplifying and pattern-perceiving heuristics, which enables them to make sense however complex the reality they face.

In this chapter, we assumed that those heuristics and flexibility in their judgments originate from their interwoven memory structures consisting of patternized event evolutions at various abstraction levels, and suggested the multi-layered architecture of knowledge-based DSS. The system was designed to realize the capacity to generate dynamic evolving patterns of events at different abstraction levels in a hierarchical multi-layered fashion above the empirical knowledge store, capturing event concepts using a meta-knowledge with respect to human social behavior.

CHAPTER 7

SHEMATA-BASED INTERFACE SYSTEM FOR DECISION SUPPORT

7.1 Introduction

Although what is stored in the originally encoded knowledge consists only of a collection of nodes and the binary relations among them, the organized conceptual hierarchy according to the procedure described in the previous chapter enables the user to gain access to the large knowledge store selectively and utilize it conceptually in retrieving knowledge. The hierarchical relationships among events are similar to the human memory structure and accordingly realize a natural conversation between the system and the user. That is, it assists his global understanding of the situation or the context of the problem, as well as his detailed investigation into it. Moreover, since the multi-layered hierarchy is organized based on the structural or syntactical properties of the causal networks apart from their semantic contents, it becomes possible to detect similarities in spite of the difference of verbal expressions. This supports the user's intellectual tasks, such as the formation of metaphors and analogies on behavioral patterns, a process commonly recognized as the key to creativity in formation of scenarios (cf. Winston, 1980, 1982).

In this chapter the hierarchical knowledge is organized as interwoven *schemata*, structures like *frames* (cf. Minsky, 1975; Bobrow and Winigrad, 1977; Roberts and Goldstein, 1977), and stored in schemata base in the system. In order to make access to the knowledge smooth and flexible, the system is facilitated with a natural language interface. The conversation with the knowledge base can be done through the schemata base by means of a number of useful commands as well as by natural language inquiries. The system is implemented by the logical programming language of PROLOG, whose power of predicate calculi and pattern matching capabilities are well utilized both in constructing hierarchical knowledge structures and in processing natural language inquiries (cf. Sawaragi, et al., 1986d, 1987b, 1987c). Fig.7.1 shows an overview of the system.

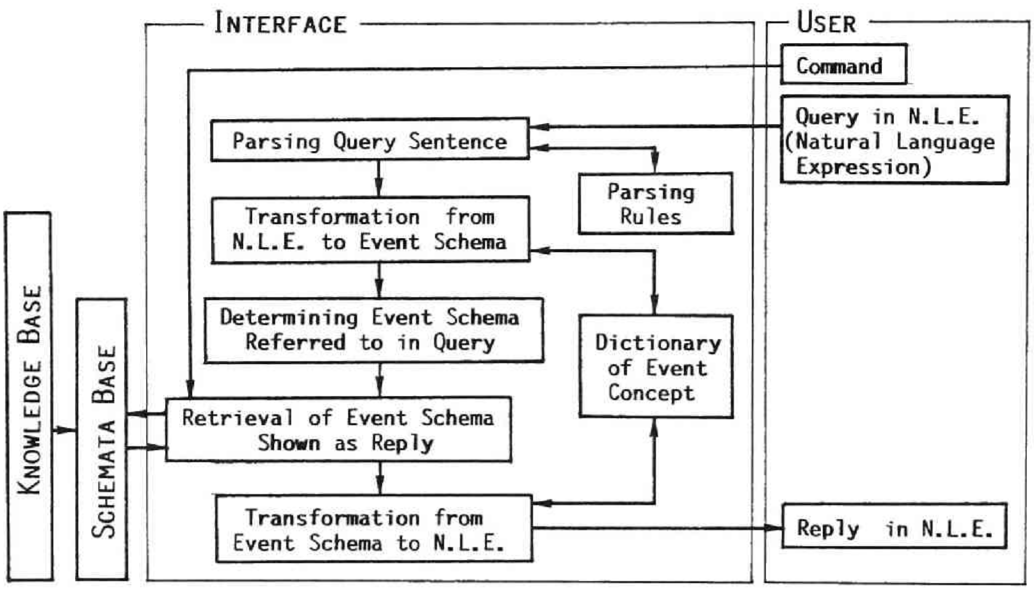


Fig. 7.1 Process flow of the interface system

7.2 Construction of Schemata Base

7.2.1 Specifying a Region of User's Interest

The bottom-level or surface-level knowledge base consists of sets of binary relations existing among P, V, and D nodes as well as C, T, +S, and -S relations among them. These are extracted manually from the documents on the problem of "Japan-US trade imbalance." The process of constructing the knowledge base is very simple. Causal Network Model (CNM) representations are extracted from each statement in the document, and if a content of the same node is referred to in different statements, the causal chain extracted from each statement is combined through this common node. Although such combinations contribute much to the saving of memory needed in the knowledge store, the network may get complex since it is interwoven with so many causal sequences. Moreover, the number of binary relations gets larger, which puts a significant burden on the user to grasp the global and essential structures of the problem as a whole. This makes it difficult for the decisionmaker to focus his interest. In order to overcome these difficulties, our suggesting method of constructing multi-layered knowledge structures above CNM representation can offer the user, i.e., the decisionmaker, kinds of facilities useful in this phase of handling such informal, descriptive, and qualitative data or knowledge.

In order to focus a region of interest from the vast causal network, we have already suggested a number of methods in Chapter 4, where the knowledge is represented in cognitive maps and the region of interest is specified as a partial causal structure consisting of nodes, i.e., concepts, satisfying some criteria designated by the user interactively. That is, at first, the user inputs either categories, key words, or source documents in which he is interested, and the skeleton map that consists of those specified nodes and represents an outline of the region is constructed and presented. Seeing this, the user then proceeds to a more detailed analysis of the problem by asking the system for a detailed hierarchical map concerning the part he wants to investigate more fully.

Such focusing methodologies can be also applied to the knowledge base of the CNM representation as well, but since this representation is characterized by its actor-orientedness and intermingles the various actors participating in the problem, the focusing process proceeds in the current version as follows. The user specifies some actors whose goal relationships he wants to explicate. In this step, two modes of specification are available.

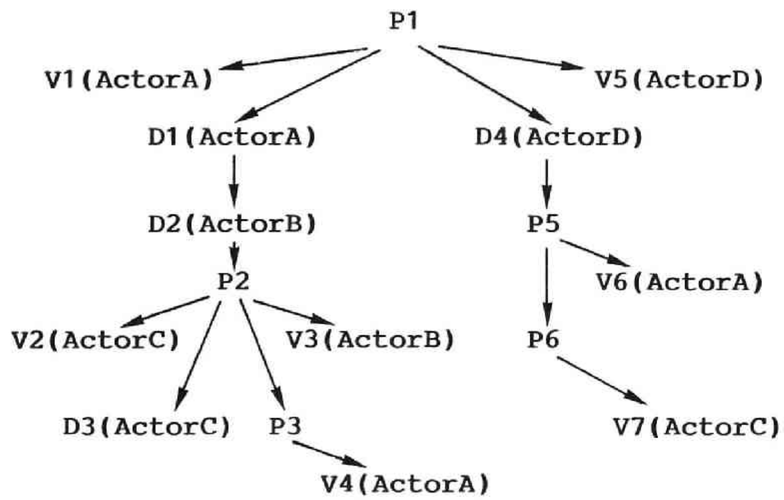
(1) Partial Specification

The specified actors are considered disjunct. The system searches the causal sequences[1] where at least one of those actors is participating, that is, the causal sequences which include the D or V nodes concerning those actors. Usually these sequences consist of nodes with respect to actors other than those the user specifies. The system searches all those nodes, and returns a list of causal sequences as well as a list of actors concerned.

Referring to those outputs, the user can grasp the interventions of even unexpected actors, and if he wants to focus on some specific actors from among them, he can use the following mode.

---

[1] The causal sequence here is actually a tree-like structure, where a tree is a series of P and D nodes with leaves of V nodes.



Partial Specification with [ActorA, ActorB]

three sequences:

```
P1 -> D1 -> D2 -> P2 -> D3 [ActorA, C, D]
P1 -> D1 -> D2 -> P2 -> P3 [ActorA, B, D]
P1 -> D4 -> P5 -> P6       [ActorA, C, D]
```

Full Specification with [ActorA, ActorB]

one sequence:

```
P1 -> D1 -> D2 -> P2 -> P3
```

Fig. 7.2 Two kinds of specification modes for the knowledge base

(2) Full Specification

In this case, the specified actors are considered conjunctive. The system searches the sequences where they all participate at the same time. Since these sequences have nodes concerning with other actors as well as the specified ones, the system excludes from the sequences D and V nodes with respect to the other actors, as well as linkages incident in and out of those nodes. This mode is useful when the problem includes too many actors and is too complicated for the user to grasp. In this case, he can filter out of the complicated reality just those aspects he is interested in. Fig.7.2 illustrates the two specification modes. As this figure shows, the specified sub-networks differ according to the selection of modes as well as the designation of actors. Those individual networks produce scenarios from different viewpoints as a sequence of event schemata.

7.2.2 Organizing Causal Networks into Multi-Layered Structure of Schemata

For all kinds of event-types at three different abstraction levels, primitive, complex, and goal interactions, the system stores their general specification forms as a *template scheme*. Each template schema has a corresponding set of attributes of events as slots, and their values. Fig.7.3 and Fig.7.4 illustrate template schemata of complex event "revenge" and primitive event "competition," respectively, with their *semantic network* representation (Quillian, 1968). Here, the following kinds of slot-value pairs are prepared; the first slot defines its *event type* "revenge" and the second and the third attributes show two *actors* A and B, and two *contents* C and D, respectively, where the numbers of actors or contents are determined according to the event-type. The schema has such kinds of pointers provided with information on its relations with other schemata as *a-part-of*, *has-part*, and *means-by* relations, each of which denotes its

```
complex event:

  event-type:     revenge
  actor:         [A, B]
  content:       [C, D]
  has-part:      [p_ev(competition),
                  p_ev(problem),
                  p_ev(success*),
                  p_ev(competition)],
  a-part-of:     <g_sq(neg_int) or g_sq(pos_int)>
  means-by:      <c_ev of consistency>
  text:         [[A,revenge,on,B,for,C,by,doing,D],
                 [A,retaliate,upon,B]]
```

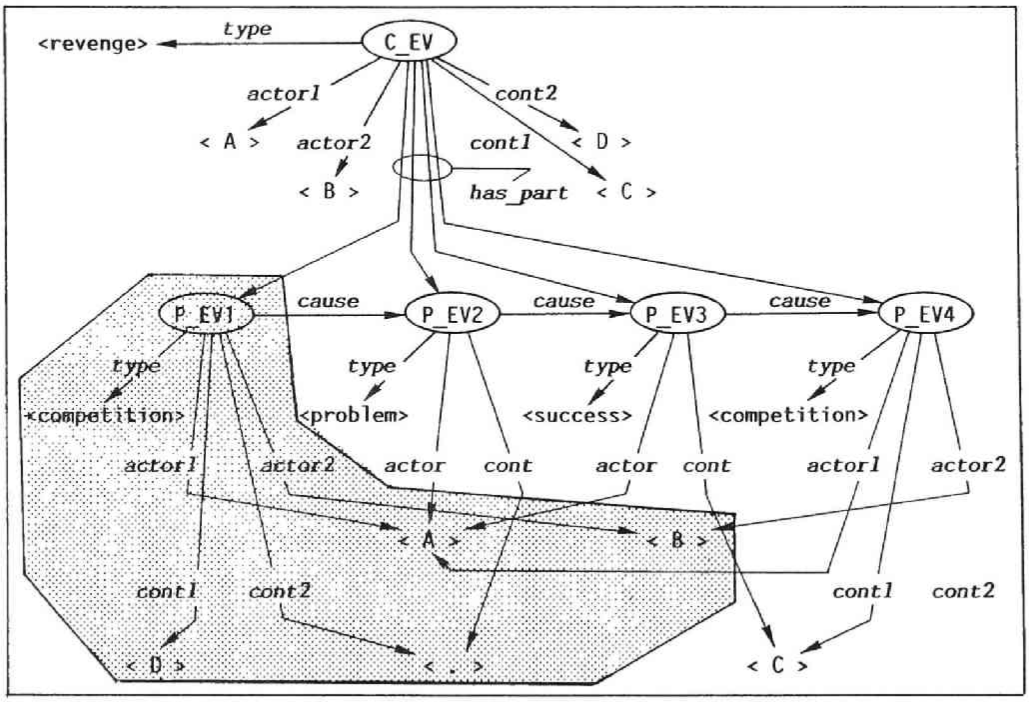


Fig. 7.3 Illustration of template schema of complex event "revenge"

```
primitive event:

  event-type:     competition
  actor:          [A, B]
  content:        [C, D]
  has-part:       [[p_node,v_node],
                    ............
                   [p_node,v_node]]
  a-part-of:      <c_ev>
  text:          [[A,compete,with,B,for,C],
                  [A,compete,with,B,for,C,and,D]]
```

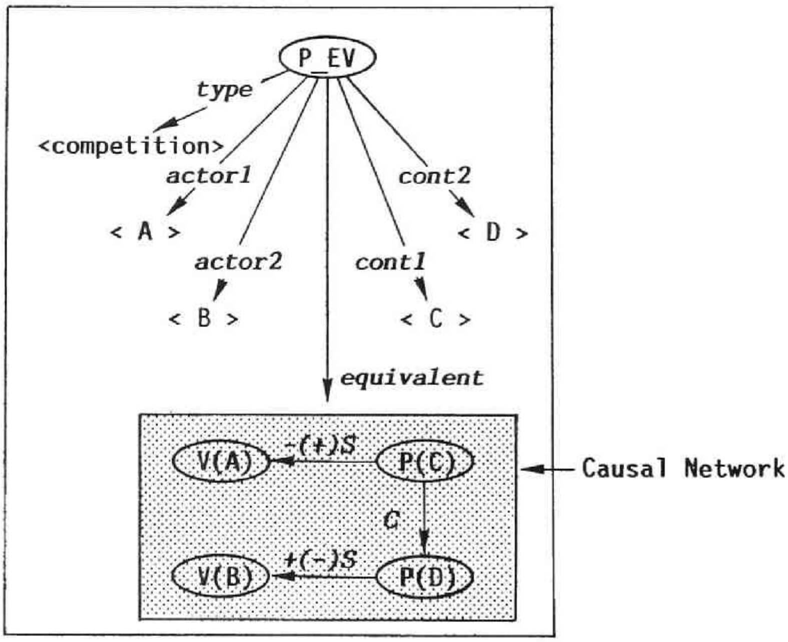


Fig. 7.4 Illustration of template schema of primitive event "competition"

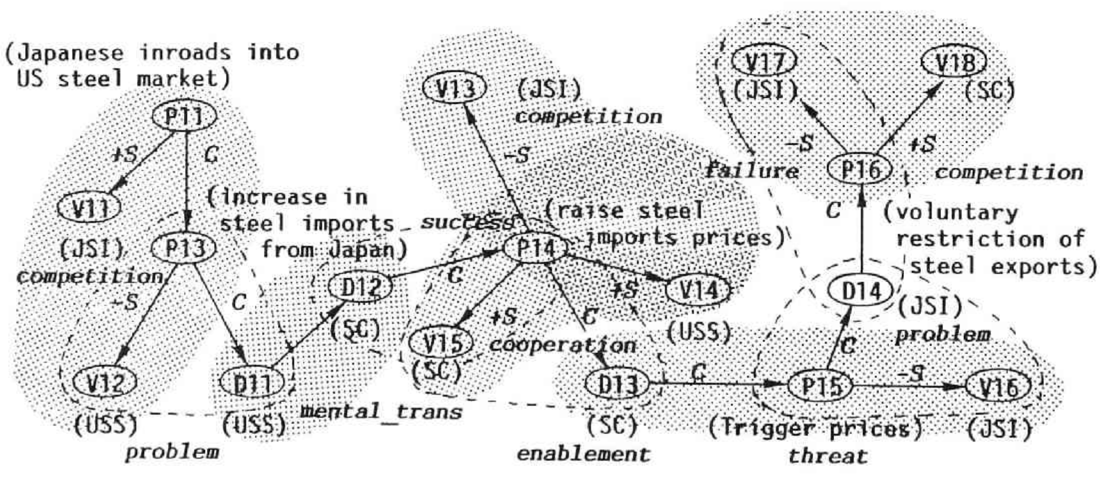
upper-level event schema, its lower-level event schemata, and its instrumental event schema, respectively. Through these pointers, the relative positions within the overall knowledge organization for each event schema is defined. The slot of *text* defines in list form the natural language expressions that can be used to represent the conceptual meaning of the event. The capital letters inserted in these bind the same entities appearing in other values denoting the correspondences between articles of natural language expressions such as subjects, verbs, and objects with the slot-values of the event schemata. Since an identical conceptual meaning of the event can be usually represented by different natural language expressions in so many ways, the values of this slot "text" can be filled with some different expressions.

On finding the event in the specified region, the system instantiates the corresponding template schema with their slot values filled with the entities found in the CNM representation. For instance, for the CNM representation of Fig.7.5(a), the same with Fig.6.5 in the previous chapter, the template schema of "revenge" is instantiated, and then the instance schema c_ev(12) shown in Fig.7.5(c) is generated. In this instance event schema, the slots of actor and content are filled with two actors [US_Steel_Co., Japan_Steel_Ind] and two contents [[Japanese,inroads,into, US,steel,market], [increase,in,steel,imports,from, Japan]], respectively. Its has-part pointer has a list of its component lower-level event schemata as its value, i.e., p_ev(14):"competition," p_ev(15):"problem," c_ev(11): "accepted_request," functioning as an instrumental means, and p_ev(16): "competition," all of which are generated as a result of instantiating their corresponding template schema. On the contrary, a-part-of pointer denotes its upper-level structure, i.e., goal interactions of g_sq(11):

"negative-interest." Its means-by pointer has a value of c_ev(11), denoting that this event is performed by making this complex event an instrumental means. All the event schemata organized from the CNM representation of Fig.7.5(a) are shown in Fig.7.5(b)-(d).

7.2.3 Schemata Base

Each instantiated instance schema is registered and accumulated in the schemata base. Although the schemata base apparently looks like an accumulation of individual event schemata, these are actually interwoven with each other through kinds of pointers and they form a hierarchy with different abstraction levels for individual episodic scenarios. Fig.7.6 illustrates the system's graphic display of such a hierarchy for



(a)

(continue to next page)

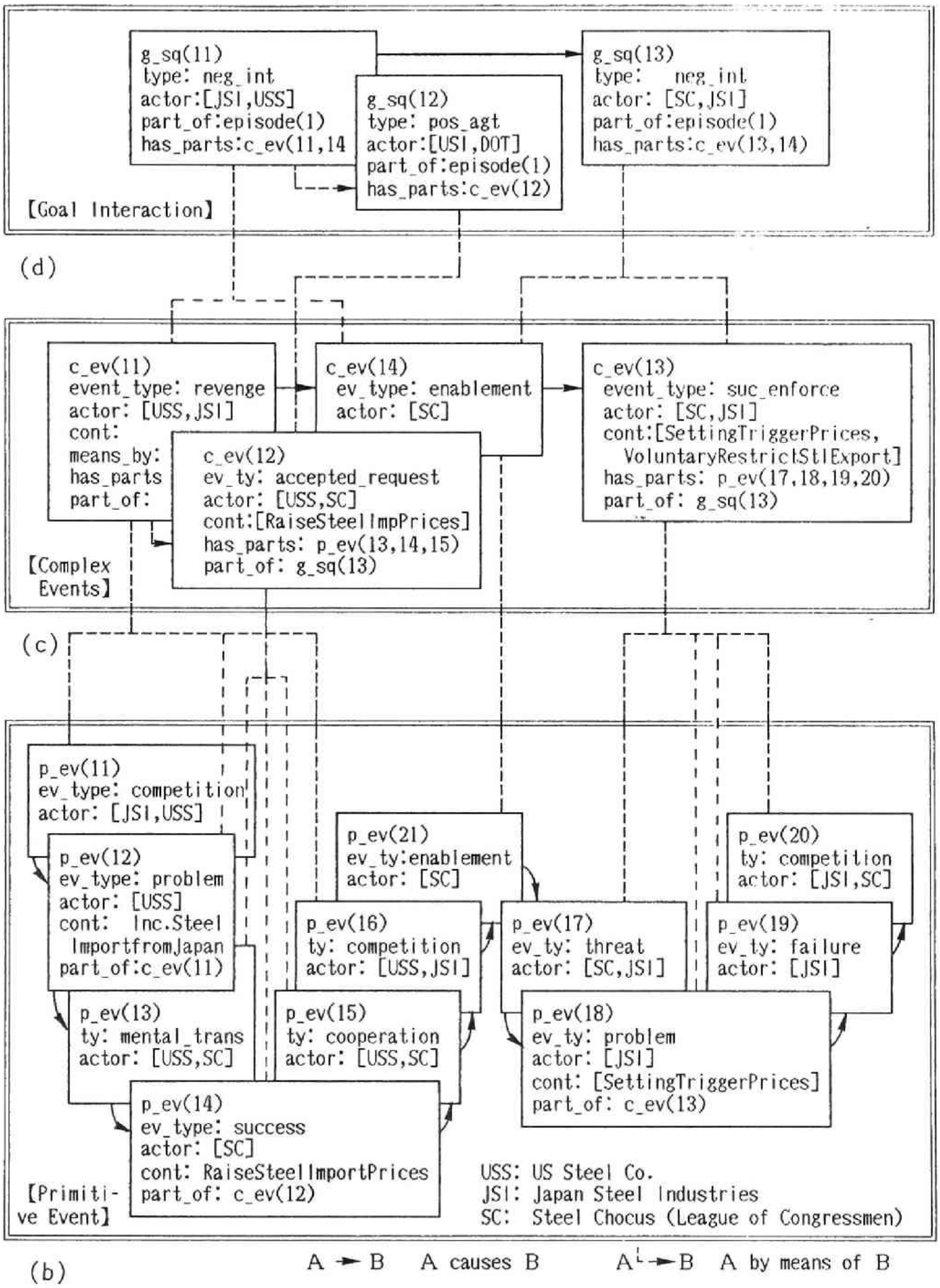


Fig. 7.5 Hierarchically organized event schemata

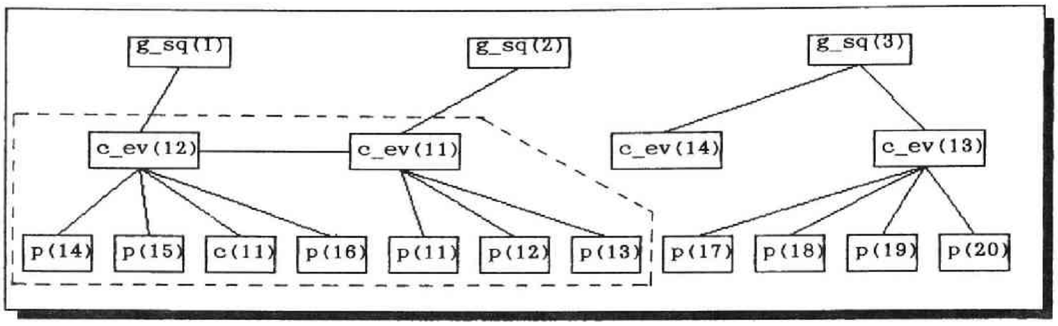


Fig. 7.6 Graphic display of schema organization

scenario(1) consisting of instantiated event schemata in the specified region of knowledge base of Fig.7.5(a). In this display, the abstraction levels from the primitive to the goal interactions change from the bottom to the top, while the events at each abstraction level evolve with the elapse of time from the left to the right. The lines connecting different abstraction levels denote a-part-of relations (or has-part relations if read in the reverse way) among schemata within the hierarchy, and the horizontal lines denote means-by relations. Each node in the hierarchy represents reference names of instantiated schemata as shown in Fig.7.5(b)-(d).

By referring to this hierarchy as well as investigating in detail inside the nodes, the user can grasp the global view of the matter of his concern. For instance, if we notice the top layer of this hierarchy, it is found that in the antecedent half of the whole structure a goal interaction of conflict is occurring between actor A(Japan_Steel_Ind) and B

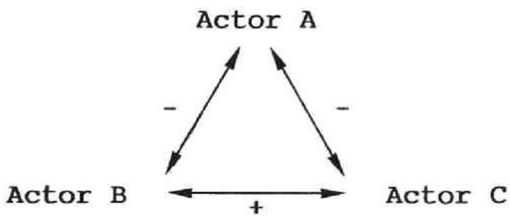


Fig. 7.7 Tripole structure implied in a part of knowledge base of Fig.7.5

(US_Steel_Co.), while at the same time another goal interaction of successful agency is observed between actor A and C(League of Steel Congressmen).

Such goal-interactions between actors have been considered theoretically under the name of *balance theory* (e.x., Heider, 1957). In this theory, a stable relationship among actors is the one that has the positive semi-cycle structure when regarding relations and actors as directed signed linkages and nodes connecting them, respectively. In our example of Fig.7.5, the extracted goal interaction relationships form a tripole structure as shown in Fig.7.7, which reveals a positive semicycle among three actors. Therefore, we can say that the goal interaction relationships are further governed by another more universal principles, that is, theory of balance.

In other words, apparently arbitrarily causally-chained phenomena represented by surface expressions encoded from documents are evolving and are controlled under the domination of a more abstract, comprehensive upper-level socio-political plan-scheme developed concurrently. Reflecting this, the constructed schemata base is a product by intermingling superficially-expressed information with socio-phenomenal commonsense knowledge. Therefore, through conversation with this schemata base, the user can be provided with kinds of eventually useful eventual information in a way compatible with his conceptual and intuitive understanding.

7.3 Interface System Based on Schemata

7.3.1 Front-End Interface between User and Schemata Base

In order for the conversation with the schemata base to go on smoothly, what is required of the interface system is a way to realize a flexible and efficient access to the enormous schemata base. Since we assume that the user does not know either what kinds of event concepts are stored or with what attributes the event in the matter is specified in the schemata base, it is not at all realistic for the user to search information based on selections among so many pre-defined menus or in a word-for-word matching way concerning with what he seeks for. Rather, as a medium to transfer his requests, it seems that the best way is to use the commonly-used terminologies and grammars that he ordinarily utilizes, that is, *natural languages*. Natural languages can exclude the burden put on the user of paying an unnecessary attention with respect to how to retrieve the information. Not only in access to the schemata base but also in receiving the system output after the system retrieves what is asked, the presentation in natural languages is much more comprehensible and understandable to the user than in the forms stored in the schemata base.

Usually in implementing such a natural language interface system, the following common problems have been brought about. First, how to organize the dictionary in a systematic way is a fundamental problem with respect to the thesauruses such as synonyms of terminologies, or to how many terminologies the dictionary is to contain to realize a sufficient communication. Second, a development of some mechanism for contexual processing will be required. That is, having retrieved some facts, the user often wants to grasp it from a more global viewpoint, or on the contrary, wants more detailed explanations with respect to it. Such a shift of the user's viewpoints from the precise, concrete level to the global, abstract levels according to the context of the conversation is one of key issues to be realized for a smooth and comprehensive communication between the system and the user.

As mentioned so far, the event schemata classify a number of different surface expressions (natural language expressions) into a definite set of conceptual event concepts, and their organization in hierarchical structures does offer significant clues for realizing those required facilities. The detailed procedures are to be described in the following.

7.3.2 Query Parsing and Generating Replies

The queries acceptable to the system in the current version are restricted to styles beginning with "Find" and its following noun clauses, that are preceded by interrogatives. Within the system, the grammar rules are defined that are required to parse the fifteen different styles of query sentences. Each parsing rule is defined according to a *Definite Clause Grammar (DCG*, in short) as shown in Fig.7.8 (Pereira and Warren, 1980). As this figure shows, in DCG each syntactical rule is defined as a horn clause of PROLOG, and the parsing proceeds utilizing efficiently the automatic backtracking capabilities of PROLOG.

As for the dictionary, it comprises frequently-utilized

```
find_np(vp(v(find),NP),SS,SE) :-    -----    (1)
   verb(find,SS,S1),
   noun_phrase(NP,S1,SE).
verb(V,SS,SE) :-
   connect_1(SS,W,SE),
   is_verb(W,_1,V),
   !.
noun_phrase(np(p(to),VP),SS,SE) :-
   prep(to,SS,S1),
   verb_phrase(VP,S1,SE).
noun_phrase(np(i(I),VP),SS,SE) :-
   inter(I,SS,S1),
   verb_phrase(VP,S1,SE).
noun_phrase(np(i(I),SNT),SS,SE) :-
   inter(I,SS,S1),
   sentence(SNT,S1,SE).
noun_phrase(np(p(P),NP),SS,SE) :-
   prep(P,SS,S1),
   noun_phrase(NP,S1,SE).
                    :
verb_phrase(vp(v(V),pp(p(P1),NP1),pp(p(P2),NP2)),SS,SE) :-
   aux_v(SS,S1),
   adverb(S1,S2),
   q_verb(V,S2,S3,P1,P2),
   prep_phrase(pp(p(P1),NP1),S3,S4),
   prep_phrase(pp(p(P2),NP2),S4,SE),
   !.
verb_phrase(vp(v(V),pp(p(P1),NP1),p(P2)),SS,SE) :-
   aux_v(SS,S1),
   adverb(S1,S2),
   q_verb(V,S2,S3,P1,P2),
   prep_phrase(pp(p(P1),NP1),S3,S4),
   prep(P2,S4,SE),
   !.
verb_phrase(vp(v(V),p(P1),pp(p(P2),NP2)),SS,SE) :-
   aux_v(SS,S1),
   adverb(S1,S2),
   q_verb(V,S2,S3,P1,P2),
   prep(P1,S3,S4),
   prep_phrase(pp(p(P2),NP2),S4,SE),
   !.
                    :
```

Fig. 7.8 Illustrations of parsing rules according to DCG

interrogatives, articles and prepositions as well as verbs, which are restricted to the ones appearing in the lists of text slots of the template schemata, as illustrated in Fig.7.9. They are represented as a collection of facts or assertions. No adjectives nor nouns are explicitly defined in the dictionary, to avoid the indefinite expansion of memory for the

```
/***PREPOSISION***/
is_prep(on).
is_prep(with).
is_prep(in).
is_prep(to).
is_prep(by).
is_prep(against).
is_prep(for).
is_prep(from).
/***INTERROGATE***/
is_inter(who).
is_inter(whom).
is_inter(what).
is_inter(when).
is_inter(why).
is_inter(how).
is_inter(whether).
/***VERB***/
is_inf_verb(causes, now, cause).
is_inf_verb(enables, now, enable).
is_inf_verb(motivates, now, motivate).
is_inf_verb(spurs, now, spur).
is_verb(spurs, now, spur).
is_verb(find, now, find).
is_verb(does, now, do).
is_verb(decides, now, decide).
is_verb(cause, inf, cause).
is_verb(find, inf, find).
is_verb(revenge, inf, revenge).
is_verb(supply, inf, supply).
is_verb(do, inf, do).
is_verb(decide, inf, decide).
is_idiom(succeeds, now, succeed, in).
is_idiom(fails, now, fail, in).
is_idiom(suffers, now, suffer, from).
is_idiom(suffer, now, suffer, from).
is_idiom(succeed, inf, succeed, in).
is_idiom(contacts, now, contact, with).
is_idiom(contact, now, contact, with).
is_q_verb(competes, now, compete, with, for).
is_q_verb(agrees, now, agree, to, with).
is_q_verb(agree, now, agree, to, with).
is_q_verb(revenges, now, revenge, on, for).
is_q_verb(recovers, now, recover, from, by).
is_q_verb(cooperates, now, cooperate, with, in).
is_q_verb(compete, inf, compete, with, for).
is_t_verb(does, now, do, against).
is_t_verb(does, now, do, for).
is_t_verb(pursues, now, pursue, for).
is_t_verb(forces, now, force, on).
is_t_verb(performs, now, perform, against).
is_t_verb(do, inf, do, against).
is_t_verb(pursue, inf, pursue, for).
```

Fig. 7.9 Illustrations of dictionary

dictionary, but a series of terms that are left unparsed as a result of applications of the parsing rules are regarded as complex nouns or noun phrases that are dealt with as a whole[2].

For instance, consider the query "Find who retaliates upon Japan Steel Industries, " shown in Fig.7.10(a). At first, this query is rewritten into the form of an ordinary list as shown in (b) detecting the space bars and periods in the query. Then, to this list, the parsing rules are applied one by one in a top down way. That is, the whole list is divided into two parts: "find" and a following noun clause according to the parsing rule (1) in Fig.7.8. Then, the latter part is further divided into interrogatives, noun phrases, and verb phrases, etc. according to the corresponding parsing rules. If inconsistency appears in the process, the system backtracks to the proper step and tries again with different parsing rules. In this way, the parsing process proceeds until the grammatical roles of each element of the list is made clear. That is, each element of the list except complex nouns matches with the dictionary satisfying their grammatical roles. For the above example, the system parses it as shown in Fig.7.10(c). After parsing the query and clarifying the grammatical roles of each element, the system rewrites the part following "find" into a list in a form of [subject, verb, object], where the query object to be asked, that is, what the interrogative ("who," in this case) indicates, is left variable as shown in Fig.7.10(d). Then, taking the match between this list

[2] The slot values of event schemata are by themselves complex nouns or noun phrases such as "Department of Treasury" and "Japanese inroads into US steel market," that are originally entities extracted from causal network model representations. Therefore, it is not required to further parse those into individual nouns or adjectives, and the matching between those chunks of terms is enough, rather than the matching in a word for word way, which makes the system performance inefficient.

```
(a)  Find who retaliates upon Japan Steel Industries.

(b) [find,who,retaliates,upon,Japan,Steel,Industries]

(c) vp(v(find),
       np(i(who),
          vp(v(retaliate),
          pp(p(upon),
             np(n([Japan,Steel,Industries],[])))))))

(d) [Var_690,retaliate,upon,[Japan,Steel,Industries]]
```

(e)

```
event-type:     revenge
actor:          [A, B]
content:        [C, D]
has-part:       [p_ev(competition),
                 p_ev(problem),
                 p_ev(success*),
                 p_ev(competition)],
a-part-of:      <g_sq(neg_int) or g_sq(pos_int)>
means-by:       <c_ev of consistency>
text:           [[A,revenge,on,B,for,C,by,doing,D],
                 [A,retaliate,upon,B]]
```

(f)

```
event-type:     revenge
actor:          [A, [Japan,Steel,Industries]]
content:        [C, D]
has-part:       [p_ev(competition),
                 p_ev(problem),
                 p_ev(success*),
                 p_ev(competition)],
a-part-of:      <g_sq(neg_int) or g_sq(pos_int)>
means-by:       <c_ev of consistency>
text:           [[A,revenge,on,B,for,C,by,doing,D],
                 [A,retaliate,upon,B]]
```

(g)

```
c_ev(12)::
 event-type:     revenge
 actor:          [[US,Steel,Co.],[Japan,Steel,Industries]]
 content:        [Japanese,inroads,into,US,Steel,market],
                 [raising,steel.imports,prices]
 has-part:       [p_ev(14),p_ev(15),c_ev(11),p_ev(16)]
 a-part-of:      [g_sq(11)]
 means-by:       [c_ev(11)]
```

```
(h) [US,Steel,Co.]
```

Fig. 7.10 Processing flow for supplementary-specification queries and the lists in the text slot of template schemata, the system determines the event concept referred to in the query and recognizes it is an instance of a template schema shown in Fig.7.10(e). At this time, other variables in the text slots are substituted into the variables in the values of other slots such as actor and content, and produce a partially specified template schema as shown in Fig.7.10(f). Then, the instance schemata of this template schema is searched that matches this partial one. After identifying it as c_ev(13):"revenge" as shown in Fig.10(g), the system returns the value of the slot (the actor slot value of c_ev(13), i.e., "US_Steel_Co.," in this case) corresponding to the unspecified variable, as an answer. If more than two instance schemata exist that match with the partial one, all the corresponding values are returned as answers.

In this way, the user can retrieve such supplementary information as who is the actor in a certain event, to whom that event is performed, or what is the content of the event, etc. by entering queries with appropriate interrogatives. Moreover, he can also retrieve all the entities in the schemata base satisfying certain conditions expressed in natural languages.

The system can also accept not only such supplementary-specification queries but a query asking to present some events from a different viewpoint. For instance, consider the query "Find why Japan_Steel_Ind. suffers from setting Trigger prices." The parsing process for this query proceeds in the same way as the former example. This time, however, the interrogative in the query is "why," asking for a more comprehensive event concept explaining the event from a more global viewpoint. That is such a dominating event concept that includes purposes or intentions of the event under inquiry. Therefore, having identified the instance schema referred to in the query, in this case p_ev(18): "problem" as shown in Fig.7.11(b), the

(a) Find why Japan Steel Industries suffer from
    setting Trigger prices.

(b)
```
p_ev(18)::
 event-type:    problem
 actor:         [Japan,Steel,Industries]
 content:       [setting,Trigger,prices],
 has-part:      [[P15,V16],D14]]
 a-part-of:     [c_ev(13)]
```

(c)
```
c_ev(13)::
 event-type:    successful_enforcement
 actor:         [[league,of,steel,congressmen],
                 [Japan,Steel,Industries]]
 content:       [[setting,Trigger,prices],
                 [voluntary,restriction,of,
                              steel,exports]]
 has-part:      [p_ev(17),p_ev(18),p_ev(19),p_ev(20)]
 a-part-of:     [g_sq(12)]
```

(d)
```
c_ev::
 event-type:    successful_enforcement
 actor:         [A, B]
                [C, D]
 has-part:      [p_ev(threat),
                 p_ev(problem),
                 p_ev(failure),
                 p_ev(competition)]
 a-part-of:     <g_sq(neg_int) or g_sq(pos_int)>
 text:          [[A,force,D,on,B],
                 [A,perfom,C,against,B]]
```

(e) Because league of steel congressmen force voluntary restrictions of steel exports on Japan Steel Industries.

Because league of steel congressmen perform setting Trigger prices against Japan Steel Industries.

Fig. 7.11 Processing flow for explanation-request queries system climbs up in the multi-layered knowledge structure in the schemata base through an a_part_of pointer of this schema and finds its upper level instance schema, c_ev(13):"successful_enforcement," as a reply to the query as shown in Fig.7.11(c).

In presenting the instance schema to the user, the system follows in a reverse way the process in which the query sentence is transformed into the instance schema. Namely, having identified the instance schema for output, the system refers to its template schema in its text slot and substitutes the variables with the values of the instance schema generating natural language expressions as shown in Fig.7.11(d), where two expressions are generated since the text slot of the schema "successful_enforcement" is filled with two different surface expression for this event concept. Finally, these expressions are printed out with the preceding conjunction "Because" and with insertions of space bars between terms as shown in Fig.7.11(e).

For queries with "how" asking more detailed explanations of some events, the system goes downwards in the hierarchy through its has-part pointers and identifies its destination, i.e., the lower level instance schemata. The individual event schemata are transformed into their corresponding natural language expressions in the same way as in the case of "why" queries, and presented to the user.

For queries with "when" asking the timing of some event, the system finds the preceding event in the horizontal causal order of event sequence in the same abstraction level by climbing up and going down through a-part-of and has-part pointers from the identified instance schema referred to in the query.

7.3.3 Summarizations and Analogizing

The user can retrieve any events with whatever degree of specification he desires in natural language expressions either directly or by climbing up and going down within the hierarchy. The event identified in this way works as a trigger to focus the user's interest, out of a vast amount of knowledge base. This focused region becomes clearer by further investigation into the peripheral events around the identified event. Since the procedures for retrieving peripheral information can be routinized and should be available to the user whenever he wants, the following commands are provided in our system.

(1) AS(Antecedent Summarization) and CS(Consequent Summarization)

Since each extracted event schema can be regarded as a plot unit consisting of the event flows at various abstraction levels implied in the encoded CNM representations, the series of event schemata can be regarded as summary of the contents of CNM representations. For this purpose, the commands of CS and AS are available in the system. When the system receives AS (or CS) command after specifying some event, which we call an *event in current focus* hereafter, through queries as described in the previous section, the system retrieves the preceding (or following) sequence of event schemata in the layer which the event in current focus belongs to. Then, those constituent event schemata of the retrieved sequences are transformed into natural language expressions by referring to their corresponding template schemata in the same way as mentioned in the previous section.

Since a degree of details of those summaries depends on the layer that the event in current focus belongs to, the user can have summarizations at desired details levels on his needs.

(2) ANG (Analogizing)

The foregoing retrieval modes are the ones concerned with how to retrieve the event schemata that are mutually connected through a-part-of or has-parts relations within the hierarchy. That is, the retrieved event schemata are the ones having either causal connections in the heterarchical meaning or predominant relations in a hierarchical sense. However, what plays a great role in stimulating the decisionmaker's creativity is not only the presentation of schemata having explicit relations in the multi-layered knowledge structure, but also the ones that are metaphorical or analogical on behavioral patterns that resemble a certain situation even though they are not identical in their surface expressions.

When the system receives the command ANG, the system tries to match the event in current focus with other event schemata already stored in the schemata base that have the same value of event type slots but may have different values in other slots. Those event schemata are transformed into natural languages as well and presented to the user.

This command can be used not only for an individual event but also for a sequence of events when it is used after the summarization commands. More precise investigations and comparisons on the similarity or difference among those analogical, metaphorical scenarios would help the user to recognize the general policies and principles that guide his behavior in a certain situation. This recognition might be a key to creativity in the formation of scenarios.

7.4 An Illustration of a Conversation between the System and the User

7.4.1 Available Information of the System

Fig.7.12 summarizes the ways in which the system retrieves the event schemata in the schemata base in reply to kinds of queries and command inputs. As this figure illustrates, the contents of the knowledge base, which is originally a collection of nodes and the binary relations among them, are presented to the user from various viewpoints on the basis of the constructed multi-layered knowledge structure moving up and down within the hierarchy, while the command ANG plays an important role in associating different, apparently independent parts of the knowledge base with each other.

Fig.7.13(a)-(d) illustrates copies of a CRT display at phases of the conversation concerned with the partial knowledge structure of Fig.7.5. As this figure shows, the conversation proceeds referring to the display with multiple windows. Those windows are: Window(1) for showing the user's inputs and the system's final outputs (Fig.7.13(a)), Window(2) for presenting the parsed result of the query and the partially specified event schema referred to in the query (Fig.7.13(b)), Window(3) for showing the instant event schema in the schemata base that matches with the one referred to in the query (Fig.7.13(c)), and Window(4) for displaying the multi-layered knowledge structure containing the event schema shown in Window(3) (Fig.7.13(d)).

7.4.2 An Illustration of Interactive Conversation

Fig.7.14(a) illustrates an interactive conversation with the interface system concerned with the partial knowledge structure of Fig.7.5 and its

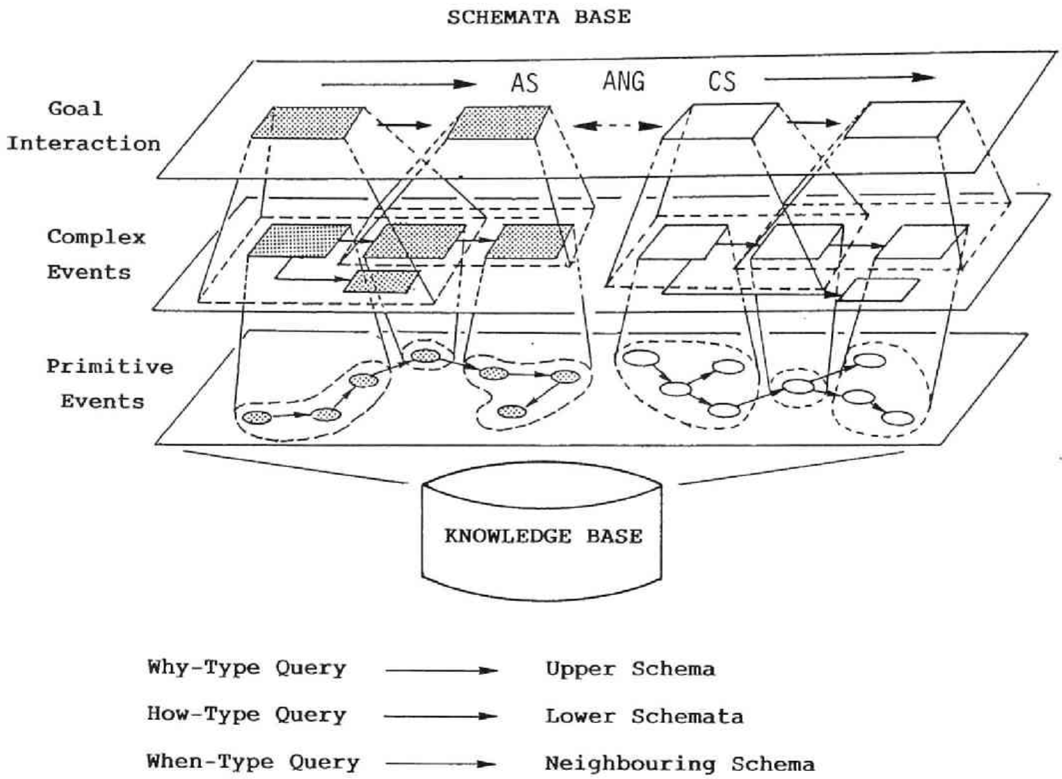
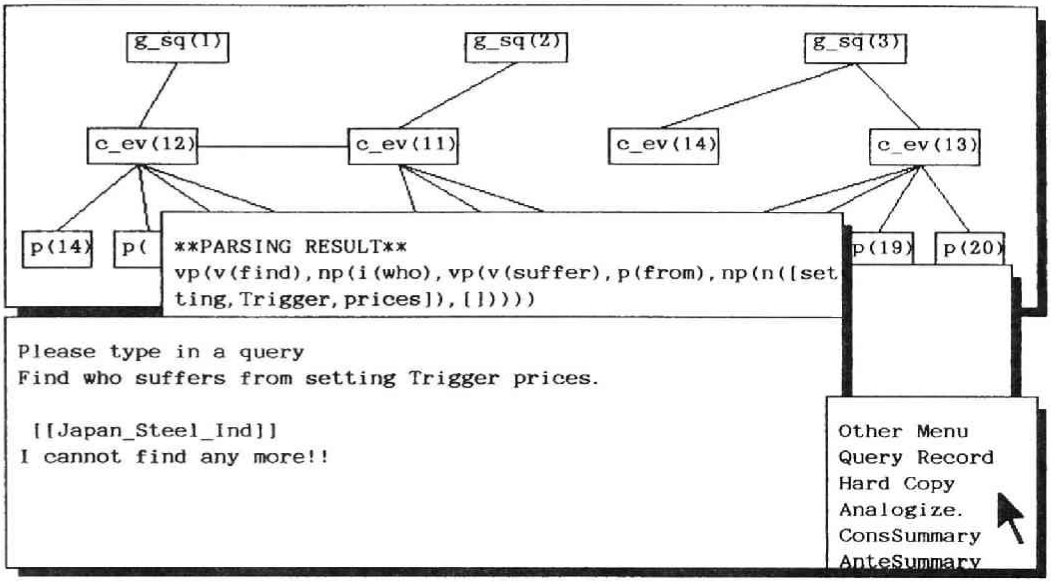


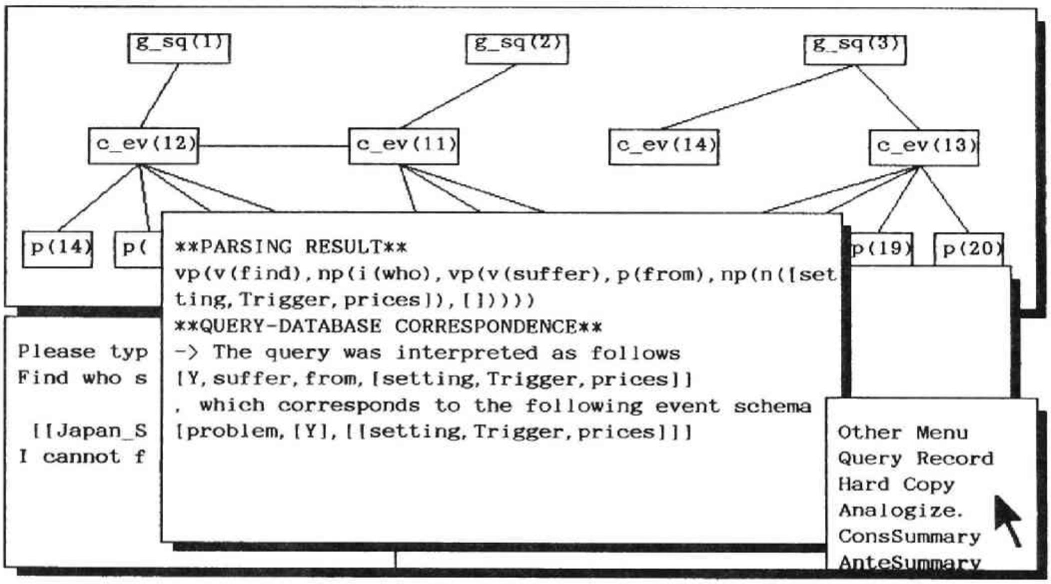
Fig. 7.12 Overview of focus-shifts within/between hierarchies of the schemata base

related parts.

In Query 1, the user wants to specify an actor who accepted US steel industries' petition of raising steel import prices. The system retrieves the event schema:c_ev(11) and returns an actor:"League of Steel Congressmen" as shown in Reply 1.



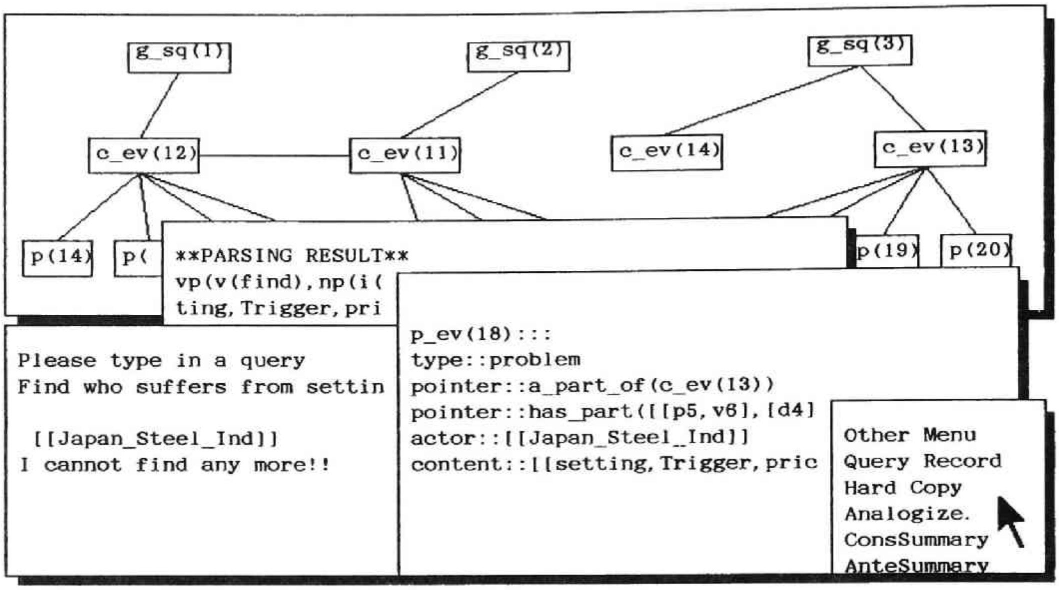
(a)



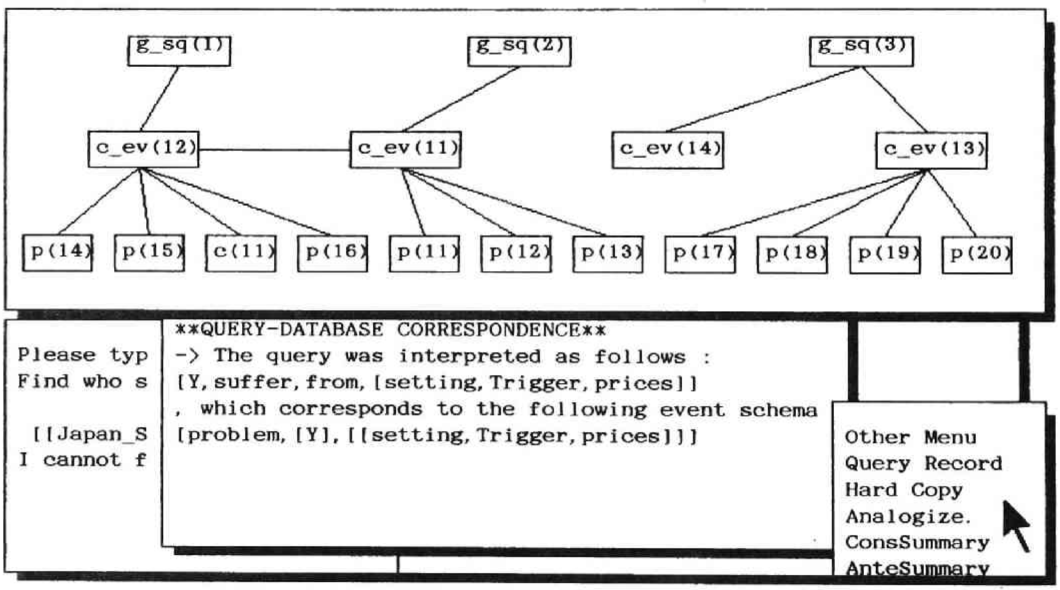
(b)

(continue to next page)

Fig. 7.13 Illustrations of CRT displays with multiple windows


g_sq(1)
g_sq(2)
g_sq(3)
c_ev(12)
c_ev(11)
c_ev(14)
c_ev(13)
p(14)
p(19)
p(20)
PARSING RESULT
vp(v(find),np(i(
ting,Trigger,pri
Please type in a query
Find who suffers from settin
[[Japan_Steel_Ind]]
I cannot find any more!!
p_ev(18):::
type::problem
pointer::a_part_of(c_ev(13))
pointer::has_part([[p5,v6],[d4]
actor::[[Japan_Steel_Ind]]
content::[[setting,Trigger,pric
Other Menu
Query Record
Hard Copy
Analogize.
ConsSummary
AnteSummary

(c)


g_sq(1)
g_sq(2)
g_sq(3)
c_ev(12)
c_ev(11)
c_ev(14)
c_ev(13)
p(14)
p(15)
c(11)
p(16)
p(11)
p(12)
p(13)
p(17)
p(18)
p(19)
p(20)
Please typ
Find who s
[[Japan_S
I cannot f
QUERY-DATABASE CORRESPONDENCE
-> The query was interpreted as follows :
[Y,suffer,from,[setting,Trigger,prices]]
, which corresponds to the following event schema
[problem,[Y],[[setting,Trigger,prices]]]
Other Menu
Query Record
Hard Copy
Analogize.
ConsSummary
AnteSummary

(d)

```
>  Find who agrees to raising steel import prices with US Steel
      Co.
                                                        (Query 1)
# [league,of,steel,congressmen]
                                 -- c_ev(11):accepted_request --
                                                        (Reply 1)
>  Find why US Steel Co. asks league of steel congressmen for
      raising steel import prices.
                                   *                    (Query 2)
#  Because US Steel Co. revenge on Japan Steel Industries for
      Japanese inroads into US steel market by doing raising
      steel import prices.
                                        -- c_ev(12):revenge --

>  ; (another expression)

#  Because US Steel Co. retaliate upon Japan Steel Industries.
                                        -- c_ev(12):revenge --
                                                        (Reply 2)

>  ANG
                                                        (Query 3)
#  UAW revenge  on Japan Automobile Industries for increase in
      unemployment of automobile workers by doing campaign for
      import restrictions against Japan.
                                        -- c_ev(22):revenge --
                                                        (Reply 3)
>  Find how UAW revenges on Japan Automobile Industries for
      increase in unemployment of automobile workers by doing
      campaign for import restrictions against Japan.
                                                        (Query 4)

#  UAW compete with Japan Automobile Industries for increase in
      Japanese automobile exports to US.
                                    -- p_ev(21):competition --
   UAW suffer from increase in unemployment of automobile
      workers.
                                        -- p_ev(22):problem --
   UAW succeed in campaign for import restriction against Japan.
                                        -- p_ev(23):success --
   Japan Automobile Industries compete with UAW for campaign for
      import restrictions against Japan.
                                    -- p_ev(24):competition --
                                                        (Reply 4)
>  CS
                                                        (Query 5)
#  UAW compete with Japan Automobile Industries for increase in
      Japanese automobile exports to US.
   Then, UAW suffer from increase in unemployment of automobile
      workers.
   Then, UAW succeed in campaign for import restriction against
      Japan.
   Then, Japan Automobile Industries compete with UAW for
      campaign for import restrictions against Japan.
```

(continue to next page)

```
   Then, campaign for import restrictions against Japan cause UAW
      to decide.
   Then, UAW contact with Japan Automobile Industries.
   Then, Japan Steel Industries suceed in luring of automobile
      factory into US.
   Then, Japan Steel Industries cooperate with UAW in luring of
      automobile factory into US.
                                                        (Reply 5)
> (UP)

#  ** Event in Focus is c_ev(22)

>  CS
                                                        (Query 6)
#  UAW revenge  on Japan Automobile Industries for increase in
      unemployment of automobile workers by doing campaign for
      import restrictions against Japan.
                                             -- c_ev(22):revenge --
   Then, campaign for import restrictions against Japan cause UAW
      to decide.
                                          -- c_ev(23):enablement --
   Then, Japan Automobile Industries agree to luring of
      automobile factory into US with UAW.
                                    -- c_ev(26):accepted_request --
                                                       (Reply 6a)
>  : (another candidate)

#  UAW revenge  on Japan Automobile Industries for increase in
      unemployment of automobile workers by doing campaign for
      import restrictions against Japan.
                                             -- c_ev(22):revenge --
   Then, campaign for import restrictions against Japan cause
      Senate to decide.
                                          -- c_ev(24):enablement --
   Then, Senate perform submit of automobile tariff act against
      Japan Automobile Industries.
                              -- c_ev(25):successful_enforcement --
                                                       (Reply 6b)
>  ANG
                                                        (Query 8)
#  US Banks revenge  on MIF for control of Eurobond market by
      doing campaign for reliese of Japanese financial market.
                                             -- c_ev(32):revenge --
   Then, campaign for reliese of Japanese financial market cause
      Senate to decide.
                                          -- c_ev(34):enablement --
   Then, Senate perform submit of trade act against MIF.
                              -- c_ev(33):successful_enforcement --
                                                        (Reply 7)
```

Fig. 7.14 Illustrations of interactive conversation between the system and the user for a part of knowledge base of Fig.7.5
* The system has no mechanism to detect whether the subject is plural or singular.

Here, in order to clarify the intention of the US steel industries' social act of petition, the user enters the why-type query shown in Query 2. Then, climbing up the hierarchy, the system finds the event schema c_ev(12):revenge that contains c_ev(11) as an instrumental means and shows it in two different expressions (Reply 2(a) and (b))[3].

Next, the user enters the analogizing command ANG (Query 3). Since the event of current interest is c_ev(12):revenge, the system tries to find a event schema of another episodic scenario in the schemata base whose event type is "revenge." The system retrieves c_ev(22):revenge with respect to the issue of "automobile trade imbalance" and presents it as shown in Reply 3.

The user wants to see the details of this event and enters the how-type query (Query 4). Going down the hierarchy with respect to this new scenario, the system presents the component primitive events p_ev(21)-p_ev(24) (Reply 4).

Then, to investigate into the subsequent scenario, the user enters the summarizing command CS (Query 5). Since the current level of abstraction is a primitive level, that is, a level of the preceding Query 4, the system presents all primitive events composing this episodic scenario (Reply 5).

The presented summary is too long and too detailed for the user to grasp, so the user intentionally shifts the focus to more abstract level[4],

---

[3] As mentioned in 7.2.2, an event represented by an event schema is usually expressed in different ways in natural languages. Therefore, the presentation of the event schema can be done as many ways as a number of values of a text slot in its template schemata. The system is assumed to present an event schema using a default expression which is determined for each event schemata, but the user can ask for another expression by entering a command ";".

[4] The user can move the focus of abstraction through the why- and how-type queries. Besides these, the user can arbitrarily go back to the abstraction level of the previous query by entering a command "(BACK)".

i.e., a complex level, and asks the system to show the summary in more compact way (Query 6). The system presents a summary consisting of three complex events, c_ev(22):revenge, c_ev(23):enablement and c_ev(26): accepted_request (Reply 6(a)).

Here, the user wants to see another episodic scenario preceded by the complex event c_ev(22). In reply to the user's prompt[5], the system retrieves another scenario that consists of c_ev(22):revenge, c_ev(24):enablement and c_ev(25):successful_enforcement and summarizes it as shown in Reply 6(b). Seeing these scenarios concerning with automobile issues, and comparing these with the case of steel issues of Fig.7.5, the user can notice the slight difference between the Japanese industries' responses to the occurrence of imbalance issues.

Suppose in this step that the user enters the analogizing command ANG (Query 7). Since the event in current focus is a sequence of complex events, c_ev(22), c_ev(24) and c_ev(25), the system retrieves such an episodic scenario in which the event types appeared in the same order as a sequence of c_ev(22), c_ev(24) and c_ev(25). In Reply 7, the system retrieves an episodic scenario from a field of financial imbalance issues.

In this way, the user can grasp some specific content of the knowledge base from various viewpoints by shifting the abstraction level. Moreover, the chunking process into more aggregated event units enables the user to detect the fundamental or structural difference between those episodes that are expressed in seemingly similar ways. On the contrary, the organized deep structure enables him to associate between different parts of the

[5] Usually in the schemata base, so many event schemata or sequences of event schemata exist that match with the event the user specifies. The system is assumed to present one of them that is found first, but it retrieves another matching candidate by the user's prompt command " : ".

knowledge base, whose associations would be by no means detected as far if just the surface expressions appeared in the knowledge base were concerned.

7.5 Concluding Remarks

In this chapter, based on the aggregation procedures mentioned in the previous chapter, the hierarchical knowledge was organized as interwoven schemata, and stored in schemata base in the system. In order to make access to the knowledge smooth and flexible, a natural language interface was developed, where the classification of verbal expressions into a finite number of event schemata was utilized effectively.

So far, as for abstraction, though we mainly dealt with only aggregation process from the details to the simple, there remains another important aspect of abstraction, that is, generalization from the specific to the general. Such a learning capability would help to eliminate a too restrictive view of the problem and realize a much more intelligent usage of the knowledge towards a decision support system that can plan scenarios for some specific situation as well as maintain consistency of the knowledge base. These will be discussed in the following chapter.

CHAPTER 8

PROTOTYPE FORMATION IN A COLLECTION OF INSTANCES AND QUANTIFICATION OF TOP-DOWN PREDICTION BASED ON SEMATIC FEATURES

8.1 Introduction

An ill-structured problem is defined as one putting demands on the knowledge as past experiences, where the decision criteria may be only loosely specified and are usually phrased in purely qualitative and ambiguous terms, without being provided with any clear-cut constraints that can discriminate definitely between what is good and what is bad, even though each individual instance were judged as good (positive instance) or bad (negative one) intuitively.

In such a domain, the human-expert decisionmaker is able to make accurate generalizations from a few scattered facts or to discover patterns in seemingly chaotic collections of observations, that is, he has an ability to generate "soft" hypotheses, which only hold probabilistically, and "partial" hypotheses, which account for some but not all of the facts. Having acquired such general expectational hypotheses, he could interpret new facts in terms of those and make sense out of complex, uncertain reality and focus his attention on making his sophisticated judgments. The truthfulness or plausibility of this intuitive judgment can be measured by the degree of fitness of a new fact to the general patterns.

In this chapter, to stimulate such a human-experts' hypotheses-driven decision process, we develop methodologies to detect interesting conceptual patterns as symbolic descriptions in the collections of individual observations, and we try to evaluate a membership of an arbitrary instance to the generated conceptual patterns by introducing the idea of *prototypicality* (Osherson ang Smith, 1981; Reed, 1972). The development of such machine-learning methodology will contribute much to the present knowledge engineering- or expert system-style approaches, in which constructing knowledge bases involves a tedious process of formalizing experts' knowledge and encoding it in some knowledge representation system (ex. Dietterich and Michalski, 1981, 1984, 1985; Michalski, 1980, 1983; Michalski and Stepp, 1984) (cf. Sawaragi, Iwai and Katai, 1987d).

8.2 Conventional Clustering vs. Conceptual Clustering

8.2.1 Limitations of Conventional Clustering

The process of constructing classifications is a form of "learning from observation." This form of machine learning has been systematically studied in such areas as cluster analysis and numerical taxonomy. The central notion used there for creating classes of objects is a numerical measure of similarity of objects. Classes are collections of objects whose intra-class similarity is high and inter-class similarity is low.

A measure of similarity is usually defined as a proximity measure in a multi-dimensional space spanned by selected object attributes. But the use of such numerical measures for constructing classifications often has the following disadvantages.

First, such a measure is meaningful only if the selected attributes are relevant and appropriate for describing perceived object similarity.

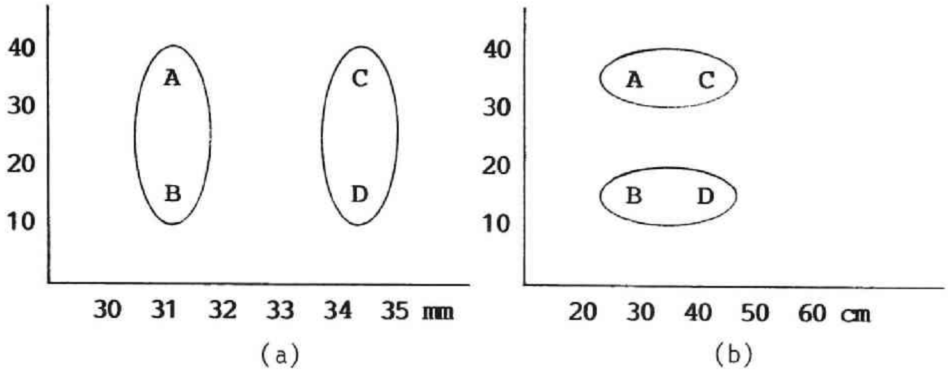


Fig. 8.1 Different clustering results according to a conventional proximity measure

The presence of irrelevant attributes will distort this measure. For instance, consider clusters formed from four objects, A, B, C, and D, that are defined with respect to two attributes, X(length) and Y(mass), as shown in Fig.8.1(a) and (b). These demonstrate that the clustering results depend on what measuring units are used to describe attributes of the objects. That is, if an attribute X is measured with a length unit of millimeters, the four objects are grouped into two clusters (A,B) and (C,D), while they are grouped into different clusters (A,C) and (B,D) when the attribute is defined with a unit of centimeters. Moreover, all attributes defining the description space are given equal weight in the process of determining classes. The problem, then, becomes one of structuring the most relevant attributes. Factor analysis and multi-dimensional scaling have been used for this purpose but these methods were designed primarily for numerical variables. They cannot adequately handle the many-valued, nominal (categorical) variables, which occur often in human classification.

Second, such measures take into consideration only the properties of the objects compared, without regard to any conceptual meaning that is useful and comprehensive for characterizing object configurations. Consequently, the resulting classes do not necessarily have any simple conceptual descriptions and may be difficult to interpret. The problem of determining the meaning of the class obtained is simply left to the researcher. This is a significant disadvantage of conventional methods because a researcher analysing data typically wants to create classes that are not only mathematically well-defined, but that also have a meaningful conceptual interpretation. Methods of conceptual data analysis are needed that generate not only mathematical formulas but logic-style descriptions, characterizing data in terms of high level, human-oriented concepts and relationships. Such derived symbolic descriptions are semantically and structurally similar to those a human-expert might produce in observing the same entities and are also easy to understand and to use for creating mental models of the information they convey.

Third, such measures of similarity are context-free, that is, the similarity between any two objects depends solely on the properties of the objects, and is not influenced by any context. Moreover, they are concept-free, that is, depending only on the properties of individual objects and not on any external or pre-acquired concepts which might be useful to characterize object configurations. Consequently, methods that use such measures are fundamentally unable to capture the properties that characterize a cluster as a whole and are not derivable from properties of individual entities. For instance, consider that objects are configurated on two-dimensionally defined space on the basis of their pairwise similarity measure as shown in Fig.8.2 (Michalski, 1983). If a conventional clustering analysis were applied to those objects, objects A and B would be grouped at first since the similarity between those is the greatest, i.e., the distance between those is the smallest, although we intuitively divide

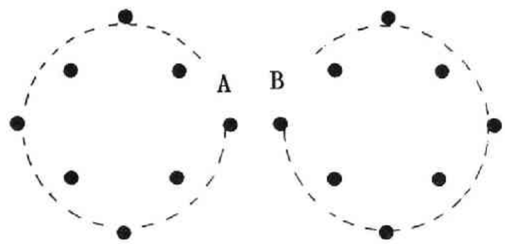


Fig. 8.2 Pair-wise similarities missing context and concepts

the set of objects into two groups as enclosed in the figure. This illustrates that conceptual clustering requires an introduction of a kind of "cohesiveness" measure which reflects not only pairwise similarity but also the relationships with the surrounding objects (context) as well as with the pre-defined concepts which we commonly have, e.g., in the example, such geometrical concepts as "diamond" and "circle."

8.2.2 Finding Generic Patterns from Chaotic Reality

Broadly speaking, the information processing activity of an organism can be characterized by self-organizing feedback mechanism. In this mechanism, the information each element presents is stored and mixed, interacting with each other, and some are diminished and others are elicited, and as a result a meaning is produced at the macro level, which returns to the individual elements and affects the performance of their self-organization. This can be interpreted as follows: the pieces of information presented, although they seemingly different from each other, have implicit connections among them. This is because they convey and detect a "meaning." Through this detected meaning, information can be united into a macro structure in its totality in some harmonious way, though each element composing it differs in some points. This structure is a product of categorization by means of meaning. This macro structure is by no means a precise one but a very crude and simple one generated from the minimum information available. However, the existence of this structure enables prompt and efficient information processing, sacrificing preciseness on the other hand, that is, processing where not all the details are not sought but just the minimum information is obtained on necessities. Moreover, this constructed macro-structure is then fed back to the newly provided information, and this is then interpreted in term of its consistency with the pre-acquired macro-structure. In this sense, the macro-structure provides with information carried by a kind of "field", i.e., a field produced by the coexistence of seemingly different individual

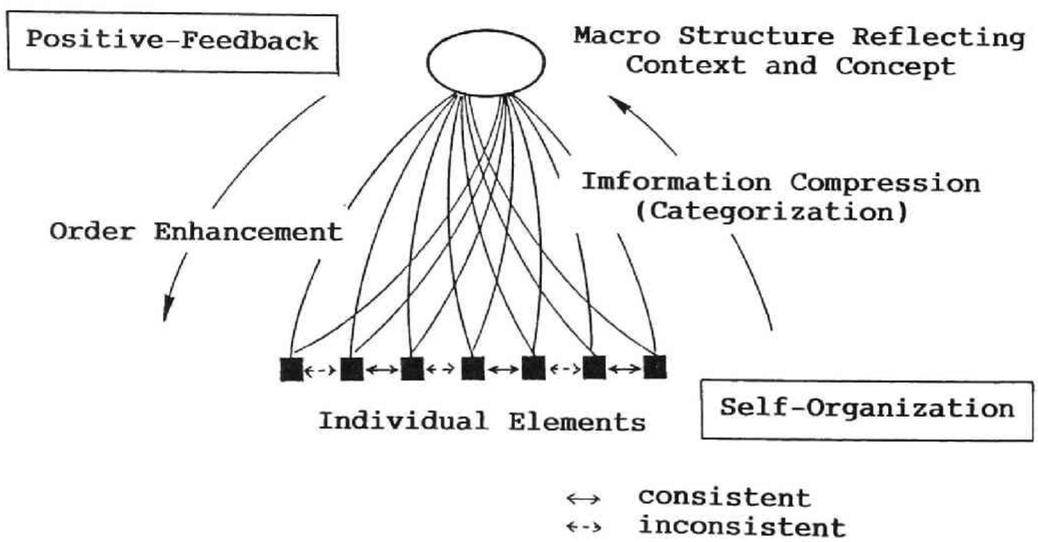


Fig. 8.3 Self-organization process according to "Holonic loop"

objects. Then, this feedback information strengthens the connectivity between the objects that are consistent with the macro-structure, while it suppresses the inconsistent connection. Fig.8.3 schematically illustrates the above mentioned processes.

8.2.3 Human Capability of Induction through Semantic Information and Prototype Theory for Concept-Driven Predictions

As for human information processing, the construction of such a macro-structure can be realized by human's inductive capability, that is, the ability of people to make accurate generalizations from a few scattered collections of observations. From the individual pieces of information, he can form such concepts that cover this information and can build an image at a higher-level than one of realistic observation.

Recently, as the theory of brain physiology develops, the function of brain with respect to its information processing has been made clear. The kinds of processes such as linguistics and logics as well as mathematically or quantitatively analytical processes have been called "left-brained" because they are associated with what are thought to be functions of the left hemisphere of the human brain. On the other hand, those tasks of a more creative, unstructured and qualitative nature, involving perceiving patterns and intuitions, have been attributed to the right side of the brain, "right-brained" capabilities (Robey and Taggart, 1982). The human capability to discriminate one piece of information from others in the macro situation, or to identify them as the same, is due to the right brain. Such an inductive learning capability can be performed by the semantics of information as well as its syntactics, which are commonly shared by people in their cultures with respect to objects.

For instance, consider a value domain for an attribute describing the

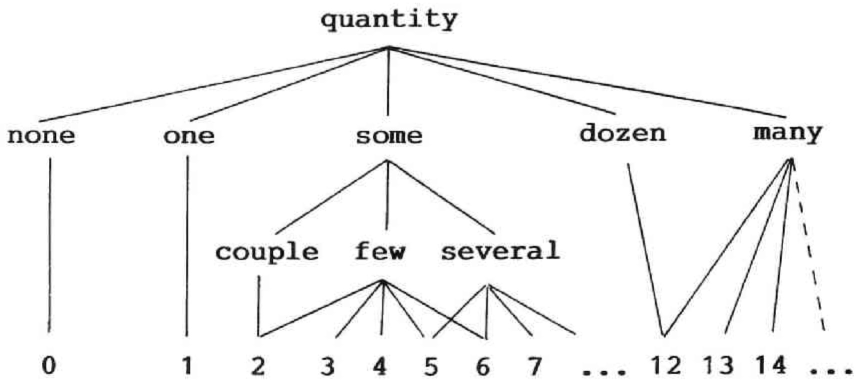


Fig. 8.4 Illustration of deep knowledge defined above continuous values

objects as shown in Fig.8.4. The lowest continuous values in the hierarchy are the ones that are really observed or measured from the objects. Indeed those values might describe the objects precisely just as they are, but when we consider grouping a lot of objects into rough clusters, those attribute values are grasped at a higher-level, i.e., in terms of qualitative, canonical or categorical values, that are illustrated above the numerical values in the figure. Moreover, when we consider such a continuous value, it depends on the domain-specific knowledge whether the observed value has a meaning because it shows that discrete value or just because that value is included in some region, i.e., a "landmark" value or a "region" value (Kuipers, 1984). Such a hierarchical structure can be by no means defined universally, but depends on the precise and careful investigation into the domain of the objects which that attribute describes. That is, even if the observed values were actually the same for two different attributes, any observed value would be interpreted

differently and the structurized hierarchy would not be identical for those attributes. For instance, some specific value may be regarded as a landmark value in some attribute, but that same value might be thought of at the same time as the one included in some region value.
Therefore, it is not plausible at all to calculate a distance as an absolute value of difference between the observed values without careful investigation of attribute-specific or domain-specific semantic meaning.

Concerning this process of categorization through semantic information, it is instructive to consider a cognitive category from the viewpoint of the *prototype theory* (Rosch, 1973, 1978; Rosch and Mervis, 1975; Rosch et al., 1976), a theory of concept formation in the psychological domain. The classical view that categories are defined by sets of necessary and sufficient features has been challenged. Even if taxonomies could agree on the unique defining features of particular objects, it seems implausible that there should exist well-articulated, all-or-nothing boundaries for common categories, but the situation may become even more fuzzy.

The newer theoretical view is that a category is defined not by absolutely critical attributes, but by *family resemblances* among its members, just as all the siblings in a family can look somewhat similar without any single feature being common to all. Experimental subjects do not necessarily agree about whether any object is definitely in a category, but there is a high reliability from groups of subjects who rate relative degree of in-ness or out-ness. This judgment refers to the so-called prototypicality of objects. Rosch and Mervis have shown that to the extent that an object's attributes are statistically common among other objects in the category (Rosch and Mervis, 1975), the object is judged more prototypical of the category.

8.2.4 An Application of Prototype Theory to Conceptual Clustering

A new clustering algorithm is developed which is based on the organismic self-organizing capability as an inductive learning through semantic information, and on the prototype theory that is used as a measure in the feedback process from the constructed macro-structure. Here, prototypicality is utilized to evaluate the harmoniousness of individual

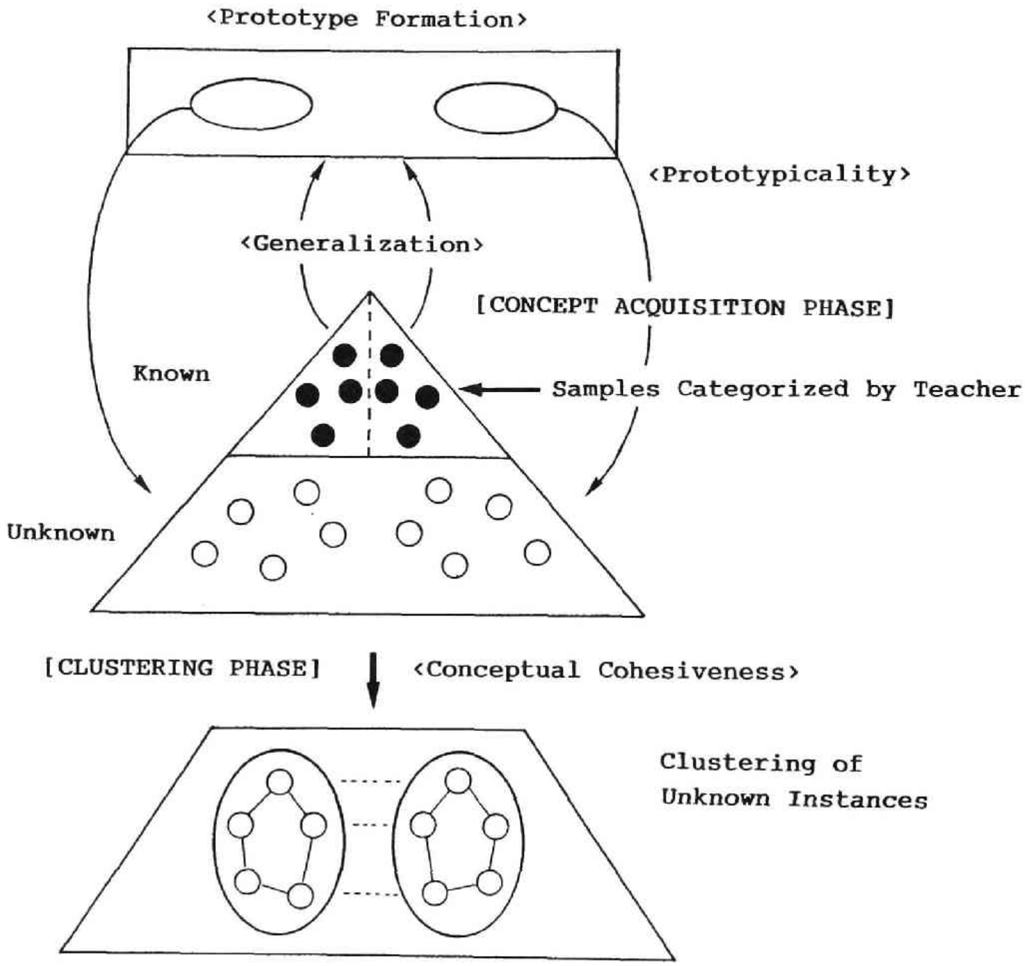


Fig. 8.5 Processing flow of the proposed conceptual clustering

elements in its totality. Fig.8.5 illustrates the procedures of the algorithm. It can be divided into two parts: the concept acquisition phase and the clustering phase.

In the former, some instances are provided by a teacher. That is, the teacher designates which instance belongs to which categories, and it does not matter whether this is done by his intuition or as a result of precise analysis. Usually these categories are competitive ones, such as normal vs. abnormal classifications, or good vs. bad categories. All the learning instances are provided as logical expressions of their composing elements, and semantic relationships such as the generalization relations among entities are also given. Using this information, similar entities in the semantic sense are connected with each other and produce more generally-expressed entities that imply these. On the basis of those generalized entity descriptions, for each category the representation called *prototype* is produced that signifies the commonly shared atmospheric characteristics at the deep conceptual level from the seemingly chaotic collections of instances. Fig.8.6 shows a set of instances of biological cells represented schematically, which Michalski used as an illustration in his developing inductive algorithm (Michalski, 1983). They are assumed to be designated by the teacher as either "normal" (DNN) or "cancerous" (DNC).

After acquiring the prototype representations for each category, the algorithm proceeds into the clustering phase. In this stage, the newly provided instances that are different from the learning instances are to be clustered into taxonomies in some hierarchical manner. Then, unlike the conventional clustering where the measure of similarity between any two objects depends on the surface properties of the object, and is not influenced by context or by any pre-defined concepts, an alternative

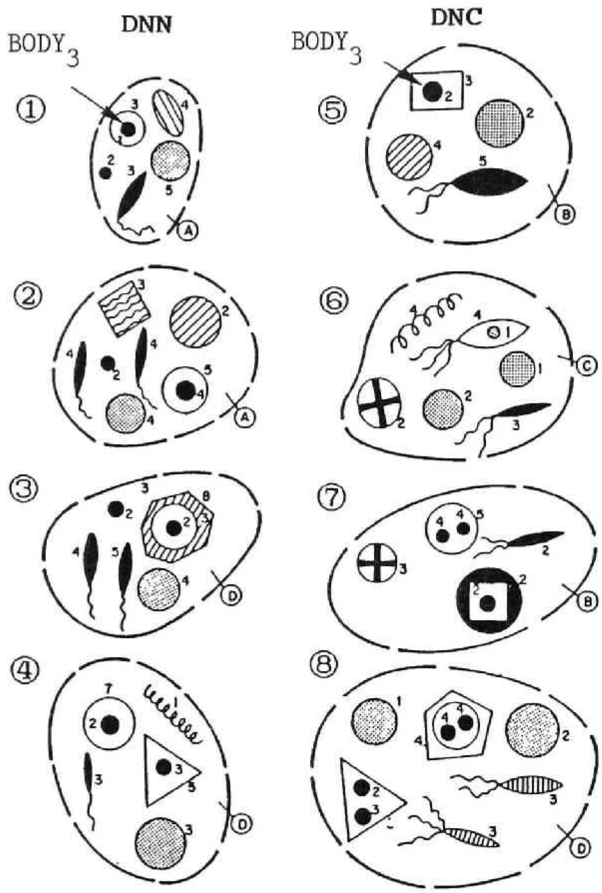


Fig. 8.6 Schematic representation of DNN and DNC cells

measure called *conceptual cohesiveness* is defined between any two objects. This is defined using the calculated prototypicality, i.e., an arbitrary instance's membership in the concepts acquired in the former phase, and designates the degree of their mutual coherence in that they can form the identical category together. Based on the cohesive matrix instead of the conventional proximity or similarity matrix, the clustering is performed. This reflects the fact that human solution involves partitioning the points into groups not on the basis of pairwise distance, but on the basis of concept membership. Points are placed in the same cluster if collectively they represent the same concept as illustrated Fig.8.2.

8.3 Prototype Learning from Instances: Case Study for Molecular Biological Field

8.3.1 Descriptive Preparation for Machine Learning

The major component of the input to a learning system is a set of statements of observed instances. Assume that an instance, $INSTANCE_i$, consists of k components, $COMP_j$ $(j=1,..,k)$ [1]. The basic forms of observed instance in our methodology are defined as follows:

$INSTANCE_i$:

<global descriptions for $INSTANCE_i$>,
<local descriptions for $COMP_1$ of $INSTANCE_i$>,
<local descriptions for $COMP_2$ of $INSTANCE_i$>,
....,
<local descriptions for $COMP_k$ of $INSTANCE_i$>,

where the parts enclosed by brackets denote a conjunctive set of *relational statements*, which are expressed in the form of [Descriptor = Value]. In the molecular biological example, an instance and a component correspond to a cell and a body (a segment composing a cell), respectively, and descriptors are divided into the following two groups:

---

[1] In this chapter, we deal with instances that consist of unordered components. That is, just an existence of the component is concerned. The learning problem from instances of ordered component will be dealt with in the following chapter.

(a) global descriptors that characterize a whole cell

. contains —— the list of bodies in the circumference of the cell

. number —— the number of bodies contained in the cell

. plasm —— the type of protoplasm in the cell (marked by enclosed capital letters in Fig.8.6)

(b) local descriptors that characterize each body

. shape —— the shape of the body

Domain: [triangle, circle, ellipse, heptagon, square, boat, spring]

. texture —— the texture of the body

Domain: [blank, shaped, solid-black, solid-grey, stripes, crossed, wavy]

. weight —— the weight of the body

Domain: [1, 2, 3,.., 5]

. orient —— the orientation of the body

Domain: [nil, N, NE, E, SE, S, SW, W, NW]

. contain_something —— the truthfulness for containment of another body inside of the body

Domain: [True, False]

. has_tail —— the body has tails

Domain: [nil, 1, 2]

. being_contained —— the body is contained in another body

Domain: [True, False],

where the value sets denoted by Domain are observable surface values to express the corresponding attributes. Note that the types of values are actually different; nominal (categorical) ones that consist of independent symbols or names, i.e., no structure is assumed to relate the values in the domain, linear ones that are totally ordered sets, and structured ones that have a tree-oriented graph structure that reflects the generalization relation between the values. In this study, such a difference is not treated explicitly but all the values are dealt with uniformly as described later.

Using the above-defined descriptions, instances provided by the teacher are described. For instance, CELL_1 in the example is represented as follows:

CELL_1:
[[contain=[BODY_1,BODY_2,BODY_3,BODY_4,BODY_5,BODY_6]],
[number=6],
[plasm=A]]

BODY_1:
<[shape=ellipse],[texture=striped],[weight=4],[orient=nw],
[contain-something=false],[has-tail=nil],
[being-contained=false]>

BODY_2:
<[shape=circle],[texture=blank],[weight=3],[orient=nil],
[contain-something=true],[has-tail=nil],
[being-contained=false]>

BODY_3:
<[shape=circle],[texture=solid-black],[weight=2],[orient=nil],
[contain-something=false],[has-tail=nil],
[being-contained=true]>

BODY_4:
<[shape=circle],[texture=solid-black],[weight=2],[orient=nil],
[contain-something=false],[has-tail=nil],
[being-contained=false]>

BODY_5:
<[shape=boat],[texture=solid-black],[weight=3],[orient=NW],
[contain-something=false],[has-tail=1],
[being-contained=false]>

BODY_6:
<[shape=circle],[texture=shaded],[weight=5],[orient=nil],
[contain-something=false],[has-tail=nil],
[being-contained=false]>,

where the sets of relational descriptions enclosed by the brackets are called *component schemata*.

The value set of such descriptors has a tree-oriented graph structure that reflects the generalization relation between the values, that is, is an *a-kind-of hierarchy*. A parent node in such a structure represents a more general concept than the concepts represented by its children nodes. In our system, the a-kind-of hierarchies are prepared for each local descriptors as shown in Fig.8.7.[2] The attached numerical tuples besides each hierarchy in the figure denote the generalization levels of the value (the more general values have the greater numbers) and the maximum depths of the hierarchies. That is, they denote the relative generalization levels of the value within the hierarchies.

8.3.2 Concept Formation from Instances

Under these descriptive preparations, the learning process proceeds as follows. At first, all the component schemata composing the provided instances are collected for each category separately. Comparing the values of the identical descriptor of the relational statements among all possible pairs of the component schemata within each category. we generate general descriptions, called *generalized component schemata*, logically implying some of the component schemata by replacing the values with more general values referring to the a-kind-of hierarchies. This process, as shown in Fig.8.8, is called the *climbing generalization tree rule*, which is formulated in principle as

---

[2] In this figure, local descriptions that have no higher-level values such as the ones with respect to 'texture' are omitted. This is also the case for 'contain_something' and 'being_contained', whose corresponding descriptors function as predicates having a value of "true" or "false."

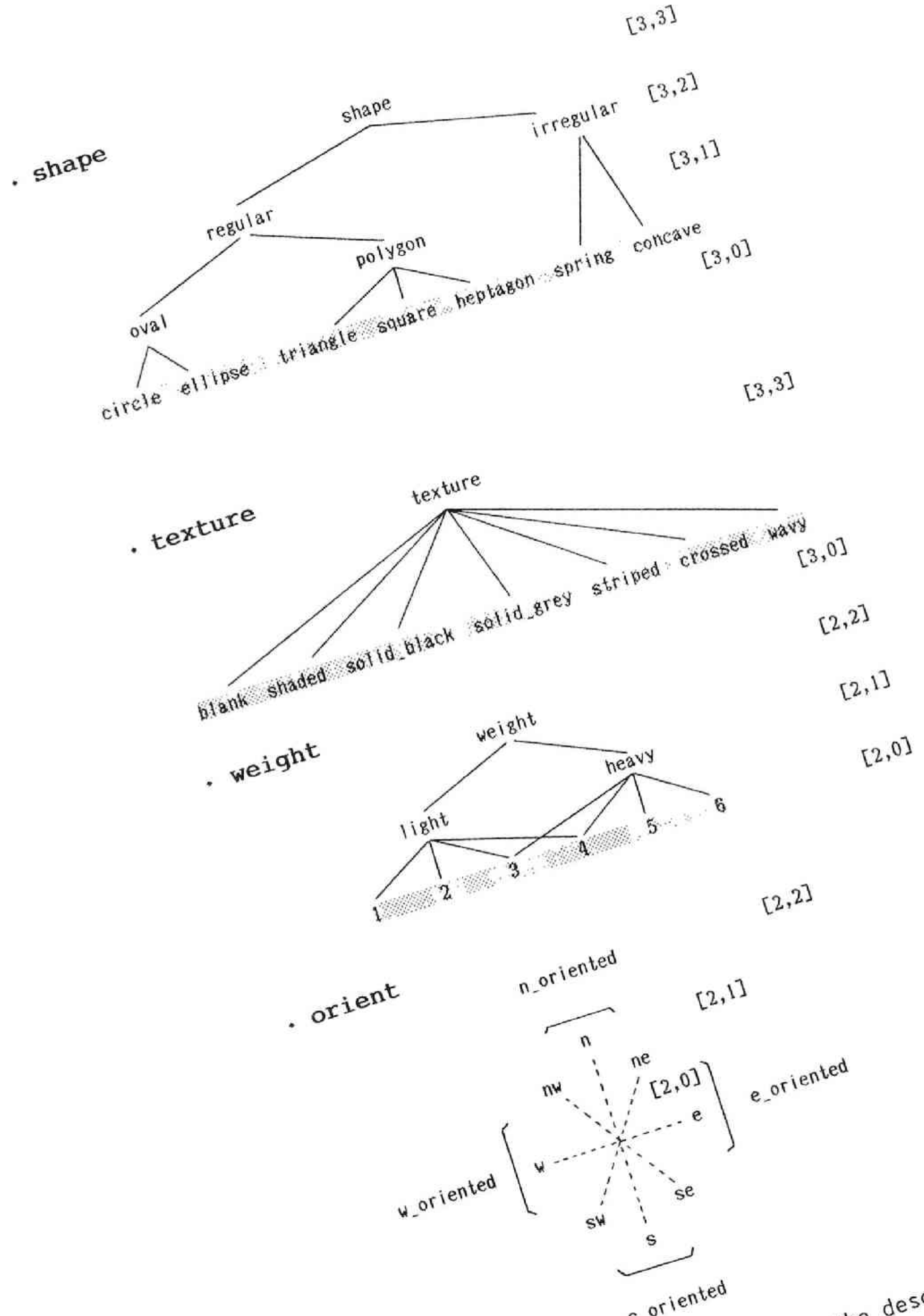


Fig. 8.7 A-kind-of hierarchies for the descriptors

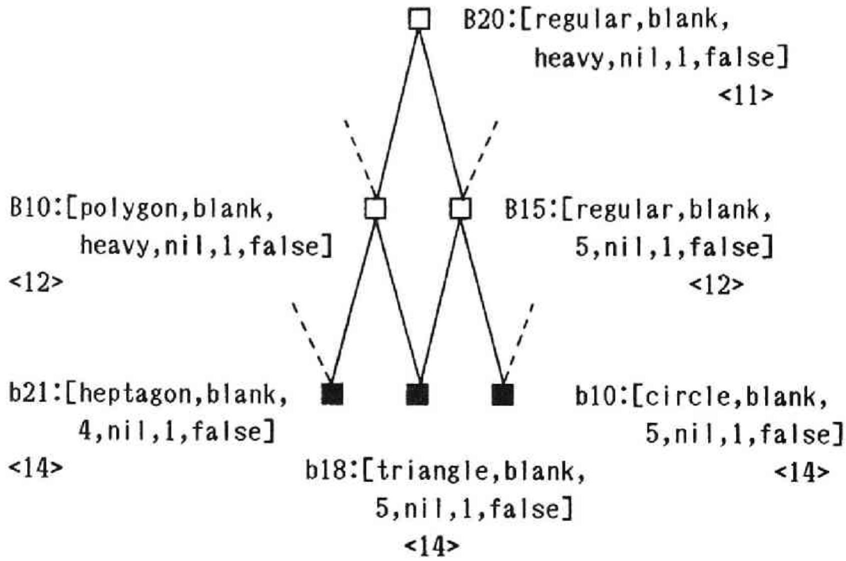


Fig. 8.8 Organizing generalized component schemata according to climbing generalization tree rules with GP ≑ 6/7

<CTX & [L = a]>
<CTX & [L = b]> (component schemata)
.....
<CTX & [L = i]>

|<(be generalized to)

<CTX & [L = s]> (generalized component schemata),

where CTX stands for some arbitrary expressions (context descriptions) that are represented as a set of relational statements and s represents the lowest parent node whose descendents include nodes a, b, ..., and i, in the generalization tree of descriptor L (the relational statement to be replaced is not necessarily unique as above). By repetitive application of this rule to all pairs of the component schemata as well as to pairs of the generated generalized component schemata, we generate the *generalized component schemata* at different levels of generalization, which produce a hierarchical tree structure relating the set-inclusion relationships among the component schemata sets implied by each generalized component schemata.

As far as general concepts are formed from observation in an inductive way, the truthfulness of the general concepts is not necessarily maintained. Especially in the above learning process, it may happen that such a schema is generated that is generalized enough to imply so many unobserved instances from a couple of observed instances. This is called *over-generalization* or *sparsed-generalization*, and is thought of as one of the criteria with which the plausibleness of the induction is evaluated.

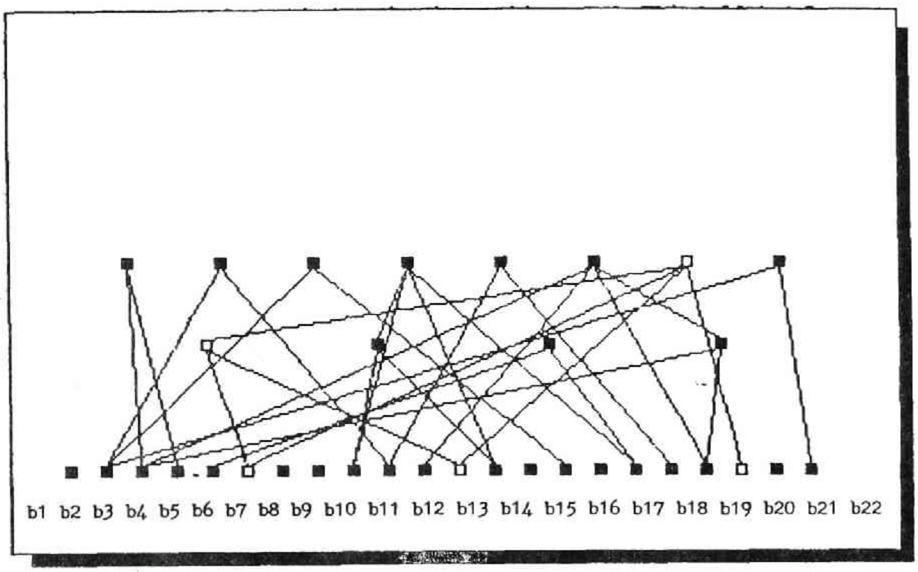
The learning procedure proceeds under the following constraints when a higher generalized component schema $E^{(h)}$ is generated by comparing a pair of lower ones, $E_1{}^{(1)}$ and $E_2{}^{(1)}$:

$$\text{concreteness}(E^{(h)}) \geq \min\left[\text{concreteness}(E_1{}^{(1)}),\ \text{concreteness}(E_2{}^{(1)})\right] \times GP,$$

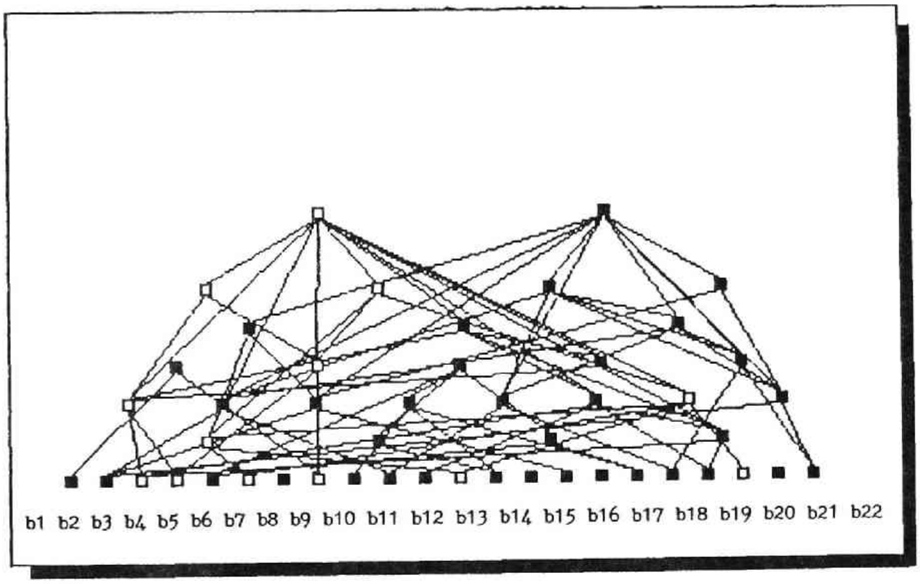
where the concreteness can be calculated from tuples denoting the generalization level of all the component descriptor values, and GP denotes a generalization parameter ($0 < GP < 1$). That is to say, in Fig.8.8 a generalized component schema E_3 cannot be generated directly from instances, $e_1 - e_3$. GP is a kind of threshold value that determines the quality of the generated concepts. When GP is set small, even such schemata as are loosely connected in their semantic meaning can be integrated into a upper generalized schema, but the generated schemata may contain such a rough one that can imply a lot of unobserved instances. On the other hand, if it is set large, just the schemata to be integrated into a generalized one are restricted to those that are sufficiently similar with each other, but the generated schema would be precise enough with fewer unobserved instances implied. Therefore, GP reflects the depth of the scope in identifying whether two different component schemata can be regarded as semantically similar ones or not, as well as an allowance of over-generalization in the learning process.

Fig.8.9 illustrates that the organized generalized schemata for the identical set of instances differ as the GP value is altered. When the GP is set smaller, the *integrability* of each schema, i.e., the degree of easiness with which each schema can be integrated into its upper generalized schema together with other schema, increases. Accordingly, the more schemata including the more generalized ones are organized above the instances, which makes the depth of the organized hierarchy the greater. Note that the organized hierarchy with some GP value, g, is a subtree structure of the one organized with smaller GP value, g'(< g). That is, the schemata (nodes) and generalization relationships among them (links) for the former are subsets of those with respect to the latter, respectively.

As for another characteristic of our suggested procedure, the organized hierarchy depends a lot on the surrounding instances or context. That is, it reflects the "field information" that is produced by being provided with the instances to learn. This is illustrated in Fig.8.10 which are hierarchies for different sets of instances, DNC and DNN, with the

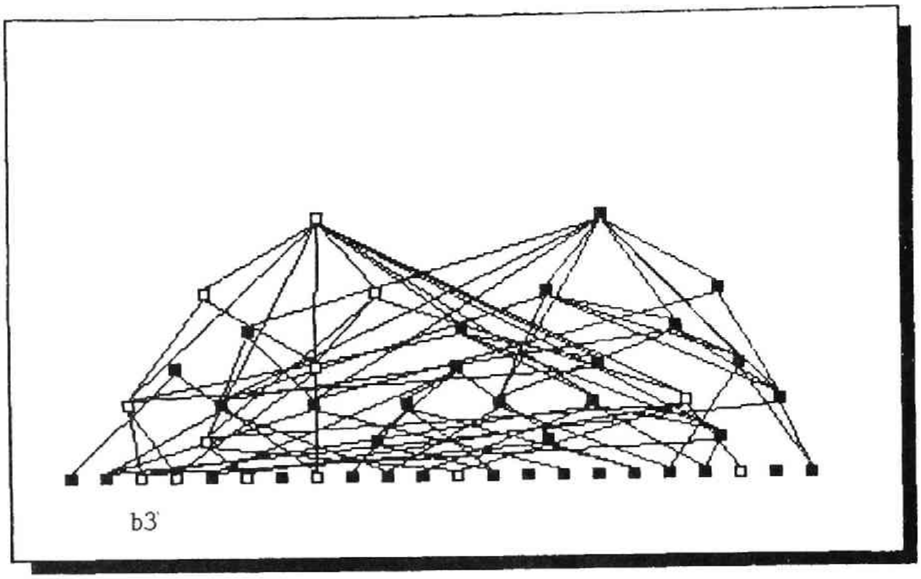


(a)

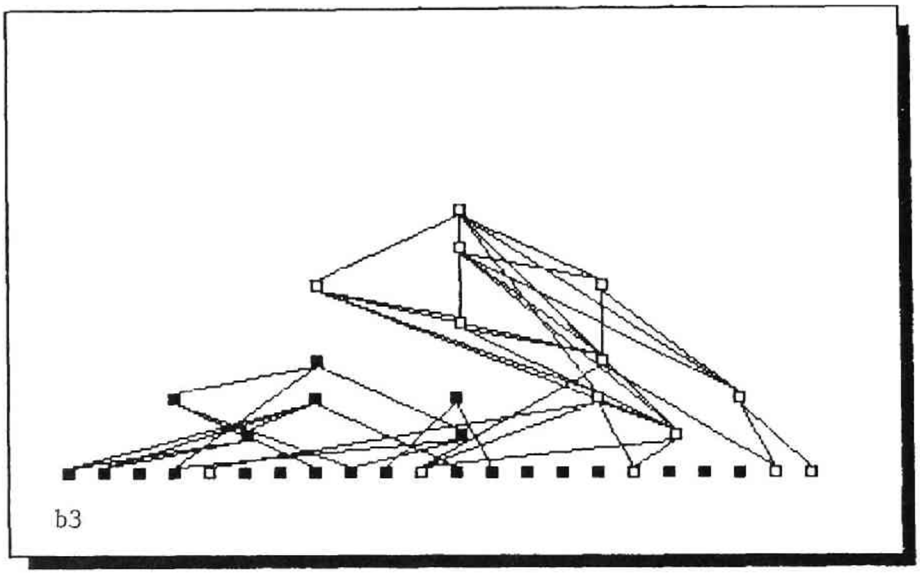


(b)

Fig. 8.9 Schemata organization with different integrabilities. (a) GP = 7/8. (b) GP = 6/8.



(a)



(b)

Fig. 8.10 Schemata organization in different context.
(a) DNN. (b) DNC.

identical GP value. Although $BODY_3$ is contained in both sets, the organized schemata above it differ greatly. In DNN, five generalized schemata, E_1–E_5, are organized, while in DNC less schemata are organized since there exists less bodies similar to $BODY_3$ in its surroundings.

8.3.2 Prototype Formation Based on Self-Organized Deep Knowledge

Then, on the basis of those organized generalized component schemata, we develop an algorithm to derive prototype descriptions of the provided collections of instances. The prototype is such a kind of mental model that represents the *central tendency* of the concept that is formed when provided with a finite set of its instances or members. It is a generalized pattern or hypothesis that accounts for the majority of the instances produced and is represented by combinations of generalized component schemata.

Based on the organized generalization hierarchies, any instance, $INSTANCE_i$, can be represented by a set of component schemata and their upper generalized component schemata together with its global descriptions, which we call a factual set for $INSTANCE_i$ and denote by $FS(INSTANCE_i)$. For instance, $CELL_2$ of DNN in Fig.8.7 can be represented by the factual set, $FS(CELL_2)$ as shown in Fig.8.11.

Then, a prototype for a category can be defined from the factual sets with respect to its members that are designated by the teacher. Here, we define a prototype description, $PD(CATEGORY_j)$, as a set of facts that are commonly shared by no less than half of the members. Fig.8.12 illustrates a prototype description of category DNN cells, PD(DNN), derived from factual sets with respect to four members, $CELL_1$ – $CELL_4$. Regarding the prototypicality of any member, $INSTANCE_i$, to a category, $CATEGORY_j$, as its

```
[number, 8, 1]
[plasma, a, 1]
[body, [circle, solid_black, 2, nil, none, f], 1]
[body, [circle, solid_black, 2, nil, none, b_cont], 1]
[body, [squre, waved, 3, nil, none, f], 1]
[body, [circle, striped, 2, nil, none, f], 1]
[body, [boat, solid_black, 4, n, none, f], 1]
[body, [boat, solid_black, 4, n, none, f], 2]
[body, [boat, solid_black, heavy, n, none, f], 1]
[body, [boat, solid_black, heavy, n, none, f], 2]
[body, [circle, blank, 5, nil, single, f], 1]
[body, [circle, blank, weight, nil, single, f], 1]
[body, [circle, solid_black, 4, nil, none, t], 1]
[body, [circle, solid_black, weight, nil, none, t], 1]
[body, [circle, shaded, 4, nil, none, f], 1]
[body, [circle, shaded, heavy, nil, none, f], 1]
```

Fig. 8.11 Factual set for $CELL_2$, $FS(CELL_2)$.

```
[plasma, a, 1]
[body, [circle, blank, weight, nil, single, f], 1]
[body, [circle, solid_black, 2, nil, none, t], 1]
[body, [circle, solid_black, 2, nil, none, f], 1]
[body, [circle, solid_black, 2, nil, none, b_cont], 1]
[body, [circle, solid_black, 2, nil, none, b_cont], 2]
[body, [boat, solid_black, 3, n_oriented, none, f], 1]
[body, [circle, shaded, heavy, nil, none, f], 1]
[body, [boat, solid_black, 4, n, none, f], 1]
[body, [boat, solid_black, heavy, n, none, f], 1]
[body, [boat, solid_black, heavy, n, none, f], 2]
[body, [circle, shaded, 4, nil, none, f], 1]
[number, 7, 1]
[plasma, d, 1]
```

Fig. 8.12 Prototype description for DNN, PD(DNN).

conceptual similarity to the prototype, it can be calculated set-theoretically using $FS(INSTANCE_i)$ and $PD(CATEGORY_j)$ as follows:

$$\mathrm{prototypicality}(INSTANCE_i | CATEGORY_j) = \frac{(FS(INSTANCE_i) \cap PD(CATEGORY_j))^{\#}}{(FS(INSTANCE_i) \cup PD(CATEGORY_j))^{\#}}, \tag{8-1}$$

$$(0.0 \leqq \mathrm{prototypicality}(INSTANCE_i | CATEGORY_j) \leqq 1.0)$$

where # denotes the number of elements of the set.

Note that once the learning of the prototype for the category is achieved, the instance whose prototypicality can be calculated is not restricted to the ones provided for learning by the teacher, but may be newly-provided or unseen instances. The more facts are commonly shared by other members the more closely prototypicality moves to 1.0, while the more facts specific to it it contains the more closely it moves to 0.0. Moreover, in case a category is formed from scattered instances that have no common facts even in the generalized level within the hierarchy, the prototypicality of all observed instances is kept smaller. Therefore, there may exist such an idealized instance that has higher prototypicality than any other observed instances provided for learning by the teacher.

The prototype can be derived taking the deep semantic information into account. Since the factual set depends on the organized hierarchy of generalized component schemata, the derived prototype may be altered according to the GP value. If GP were set to 1.0, the derived prototype would be the one based just on the surface characteristics of the instances with the deep semantic information behind it neglected.

Fig.8.13 and Table 8.1 illustrates the calculated prototypicality of cells in Fig.8.7 for two categories, DNC and DNN, with three different GP

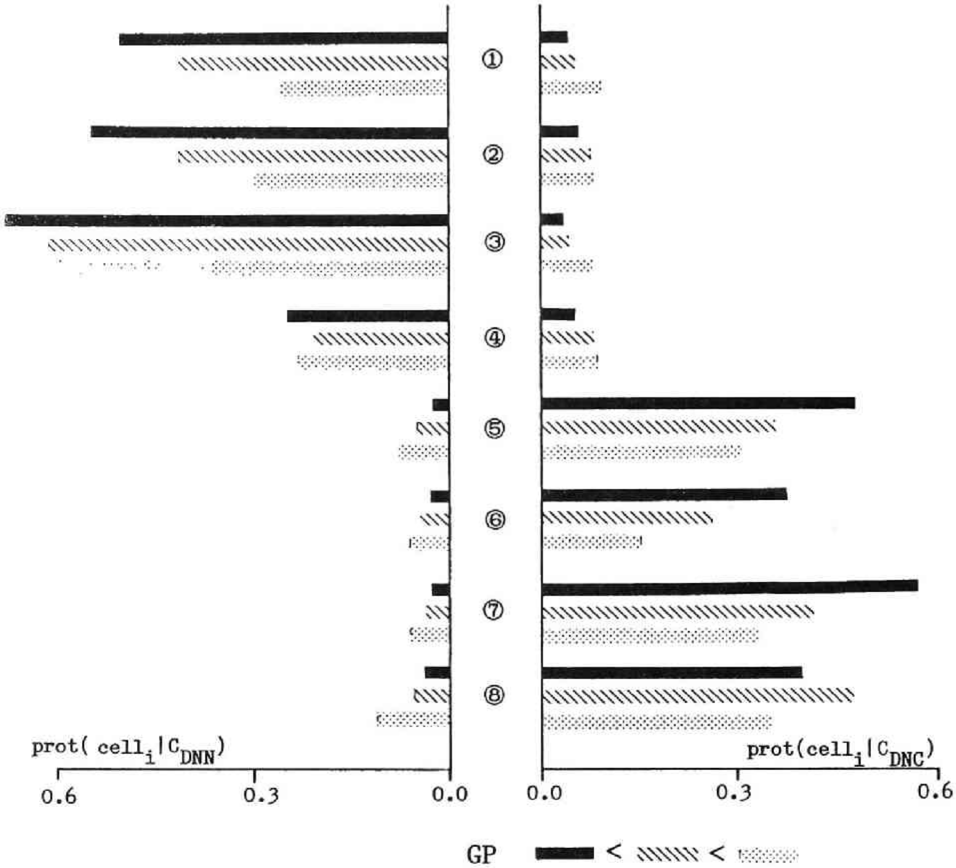


Fig. 8.13 Fuzzy sets for two categories, DNN and DNC.

	①	②	③	④	⑤	⑥	⑦	⑧
6/8	0.51	0.55	0.69	0.25	0.02	0.02	0.02	0.03
	0.02	0.05	0.02	0.05	0.47	0.37	0.56	0.39
7/8	0.42	0.42	0.61	0.21	0.04	0.03	0.03	0.05
	0.03	0.06	0.03	0.07	0.36	0.26	0.42	0.48
8/8	0.25	0.30	0.60	0.23	0.07	0.06	0.06	0.11
	0.07	0.06	0.07	0.07	0.30	0.15	0.33	0.35

Table 8.1 Calculated prototypicality for instances of Fig.8.6 with three different GP values, GP=6/8, 7/8, and 8/8.

values. These figures demonstrate that two competitive categories are formed as fuzzy sets above the identical instances. That is, any instance can have some amount of prototypicality for two categories at the same time, so that the boundary between the categories is by no means clear-cut but fuzzy. The difference of prototypicality for partially overlapping categories reflects a kind of contrast in the competitive classification situation: the greater the difference is, the more easily it can be judged intuitively as an instance of either category. It can be suggested from the calculated results of Fig.8.13 that the more the deep semantic information is taken into account, the clearer this contrast becomes.

8.4 Conceptual Clustering Based on Conceptual Cohesiveness

8.4.1 Definition of Conceptual Cohesiveness Using Prototypicality

Using the prototypicality, we define a conceptual cohesiveness between any pair of instances, which depends not only on those instances and surrounding instances, but also on a set of pre-defined concepts which are available for describing those instances together. Conceptual cohesiveness is a measure quantifying the degree to which any pair of instances can form the common pre-defined categorical concepts together. More concretely, it is defined using prototypicality as following:

$$\text{conceptual cohesiveness}(\text{INSTANCE}_i,\ \text{INSTANCE}_j) =$$
$$\max_k \min \left[\text{prototypicality}(\text{INSTANCE}_i \mid \text{CATEGORY}_k),\ \text{prototypicality}(\text{INSTANCE}_j \mid \text{CATEGORY}_k)\right] \qquad (8\text{-}2)$$

In Eq.(8-2), a minimum operator is an AND-operator and represents the conjunction of $(\text{INSTANCE}_i \in \text{CATEGORY}_k) \wedge (\text{INSTANCE}_j \in \text{CATEGORY}_k)$ in the fuzzy

	①	②	③	④	⑤	⑥	⑦	⑧
①	1.00	0.51	0.51	0.51	0.25	0.02	0.02	0.03
②	0.51	1.00	0.55	0.25	0.05	0.05	0.05	0.03
③	0.51	0.55	1.00	0.25	0.02	0.02	0.02	0.03
④	0.51	0.25	0.25	1.00	0.05	0.05	0.05	0.05
⑤	0.25	0.05	0.02	0.05	1.00	0.37	0.47	0.39
⑥	0.02	0.05	0.02	0.05	0.37	1.00	0.37	0.37
⑦	0.02	0.05	0.02	0.05	0.47	0.37	1.00	0.39
⑧	0.03	0.03	0.03	0.05	0.39	0.37	0.39	1.00

Fig. 8.14 Cohesive matrix for GP=6/8

reasoning. A maximum operator is an OR-operator and represents the disjunction of $\bigvee_k [(\mathrm{INSTANCE}_i \in \mathrm{CATEGORY}_k) \wedge (\mathrm{INSTANCE}_j \in \mathrm{CATEGORY}_k)]$. Note that although the surrounding instances are not involved explicitly in Eq.(8-2), conceptual cohesiveness implicitly reflects such contextual information, since the prototypicality is calculated based on the surrounding instances, as mentioned in the previous section.

8.4.2 Clustering Results Based on Conceptual Cohesiveness

Fig.8.14 illustrates a cohesive matrix whose (i,j) element is a conceptual cohesiveness between $\mathrm{INSTANCE}_i$ and $\mathrm{INSTANCE}_j$. Here, the conventional clustering analysis, *Weighted Pair-Group (WPG)* Method, is applied to the conceptual cohesive matrix instead of to the pair-wise similarity matrix. The results are represented in dendrograms as shown in Fig.8.15. Fig.8.16 is a conventional pair-wise similarity matrix whose (i,

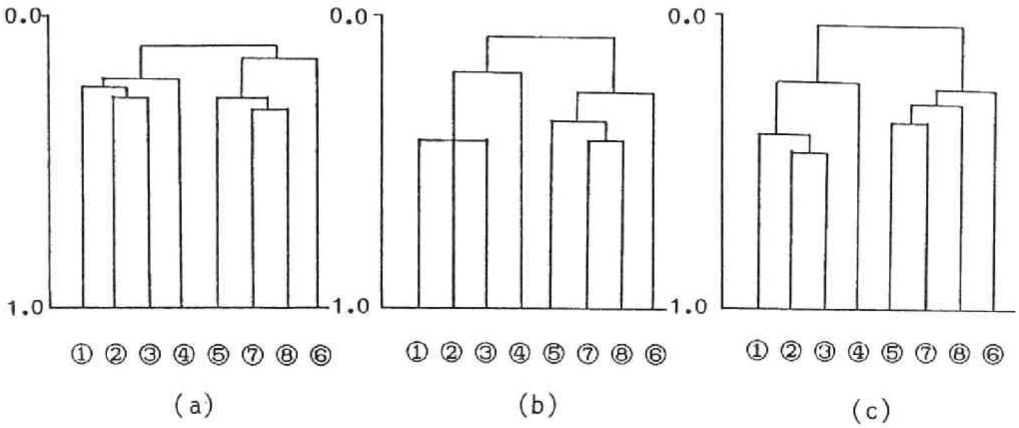


Fig. 8.15 Clustering results according to conceptual cohesiveness. (a) GP=8/8. (b) GP=7/8. (c) GP=6/8.

j) element is calculated from their surface features according to the following:

$$\text{similarity}(\text{INSTANCE}_i,\ \text{INSTANCE}_j) = \frac{(\text{FS}(\text{INSTANCE}_i) \cap \text{FS}(\text{INSTANCE}_j))^{\#}}{(\text{FS}(\text{INSTANCE}_i) \cup \text{FS}(\text{INSTANCE}_j))^{\#}}, \tag{8-3}$$

where factual sets FS's are sets of facts with respect to surface characteristics of the instances. Fig.8.16 shows that there seemingly exists no similarity among instances and there is no way to divide them into clusters.

In contrast to that, in the results of applying conceptual cohesiveness, the classification by the teacher is recovered correctly for all three different GP values. Fig.8.15 shows the difference of qualities

	①	②	③	④	⑤	⑥	⑦	⑧
①	1.00	0.12	0.13	0.06	0.07	0.00	0.05	0.05
②	0.12	1.00	0.18	0.00	0.00	0.00	0.11	0.04
③	0.13	0.18	1.00	0.20	0.06	0.05	0.05	0.10
④	0.06	0.00	0.20	1.00	0.06	0.05	0.05	0.15
⑤	0.07	0.00	0.06	0.06	1.00	0.06	0.13	0.11
⑥	0.00	0.00	0.05	0.05	0.06	1.00	0.00	0.10
⑦	0.05	0.11	0.05	0.05	0.13	0.00	1.00	0.15
⑧	0.05	0.04	0.10	0.15	0.11	0.10	0.15	1.00

Fig. 8.16 Conventional pair-wise similarity matrix

of the resulted clusters among a variety of GP values. The more the deep semantic information is considered, the better the quality is. That is, instances of the same categories have higher cohesiveness with each other, while the ones of different categories are likely to be apart , so the boundary between clusters for the competitive categories is made more discriminated.

This is schematically illustrated in Fig.8.17, which can be interpreted as follows. In a collection of observed instances, each is activated loosely through deep semantic information, which is represented by the organized hierarchies of generalized component schemata. With feedback information from the context or "field information" and acquired concepts, these activated instances get to have a tendency to select other instances through their prototypicality. That is, a kind of force is

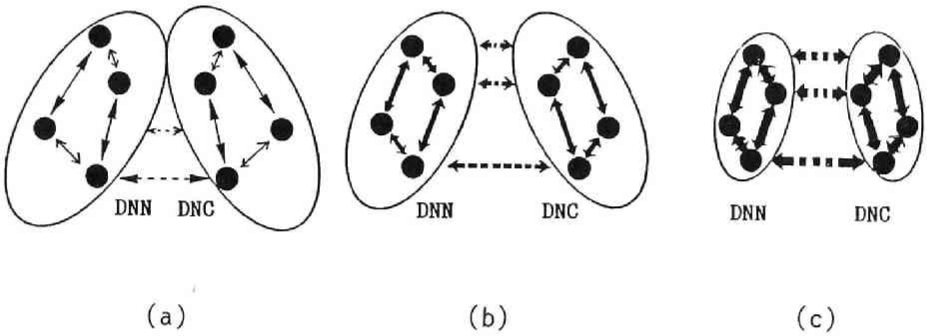


Fig. 8.17 Schematic representation of category formation of Fig.8.15. (a) GP=8/8. (b) GP=7/8. (c) GP=6/8.

generated that promotes cohesiveness among members belonging to the same category and suppresses cohesiveness with nonmembers. This force increases in proportion to the prototypicality , and as a result, the case with the greater integrability generates an order of high quality, i.e., clear division into two competitive clusters.

This may suggest a mechanism of dynamically adaptive human memory, some ideas of which have been discussed by the *connectionists* as well as in the field of brain physiology.

8.5 Concluding Remarks

In this chapter, by modelling a human-experts' capability to discover patterns in seemingly chaotic collections of observations as a hypotheses and his hypotheses-driven decision process, we developed an algorithm for conceptual clustering. Here, an idea of prototypicality was introduced, which was calculated on the basis of conceptual deep meaning behind the observed descriptions as well as of contextual information formed by a collection of instances.

We believe that our suggesting methodology is effective for such problem areas where for some instances the discrimination has been performed through an analytical investigation, but any clear-cut discriminating rules or verbal expressions have not obtained yet. To deal with such problems, it is required to investigate into a mechanism of the experts' intuitions or inductive capability from their past experiences concerning with the analytical results. Moreover, our suggesting methodology can be extendable to modelling human adaptive self-organizing capability in a dynamical way, which is a central issue of current concern among the reasearchers called the connectionists.

The development of such machine-learning methodology will contribute much to the present knowledge engineering or expert system style approaches, in which constructing knowledge bases involves a tedious process of formalizing experts' knowledge and encoding it in some knowledge representation system.

CHAPTER 9

EXPECTATION-DRIVEN DECISION SUPPORT BY LEARNING SCENARIOS FROM EPISODES

9.1 Introduction

In problem solving such complex problems as scenario formations in socio-political domains, it is seldom that an enough amount of sample data is prepared to be processed by the conventional probabilistic or statistical methods nor that any clear-cut constraints are obtained. In such ill-structured domains, empirical knowledge or episodes on individual experiences or observations play a great role rather than just the frequency or the distributions. Actually in the face of transient, unfamiliar, noisy and uncertain world reality, human cognitive limitations force decisionmakers to abandon the details and to construct simplified internalized world models at different levels of abstraction (Carrol and Payne, 1976). This makes it easy to remind them of essentially analogical case from their vast knowledge store even though it were not identical on its surface expressions. These models simplify large sets of observed facts into simple relationships and highlight the more salient variables and relationships, which are used to predict future experiences. That is, comparing a whole range of similar experiences that are to be focused adaptively depending on the current situation, expert decisionmakers derive somehow the "essence" of these experiences and generate a general expectation as a hypothesis on what happens under the situation and how the world evolves as a typical norm. Then, driven by such acquired expectation hypothesis, they can make sense out of complex, uncertain reality and focus his attention towards his sophisticated judgment. Wherein, truthness or plausibility of this intuitive judgment can be considered as a degree of fitness to that expectation structure. Such a human cognitive process or a human information process has close relationships with experts' "heuristics" that are originally psychological properties as a perceiving, thinking, and learning animal.

What is required of a DSS interfacing between those problems and the decisionmaker's judgment is to incorporate such a cognitive process that can explain how a variety of heuristics are produced, and to support their causal interpretations even for a novel case by self-organizing generally applicable knowledge based on a vast amount of accumulations of episodic knowledge.

In the preceding chapters in this Part III, we have developed a methodology to aggregate a documented data on some specific episodes into causally interrelated eventual units in a hierarchical way. Moreover, we have introduced an idea of prototypicality as an evaluating measure of pattern-directed prediction, where patterns are learned from instances as prototypes. On the basis of these, in this chapter, we summarize about "heuristics" in decisionmaking and try to model a human cognitive process that can explain how these heuristics are brought about. Then, as its application, we suggest an architecture of an interactive DSS. For this purpose, we introduce an idea of *scripts* (e.x., Schank and Abelson, 1977; Abelson, 1981). A script is a kind of psychological schemata in a form of a consistent, coherent pattern as a series of events linked together in a

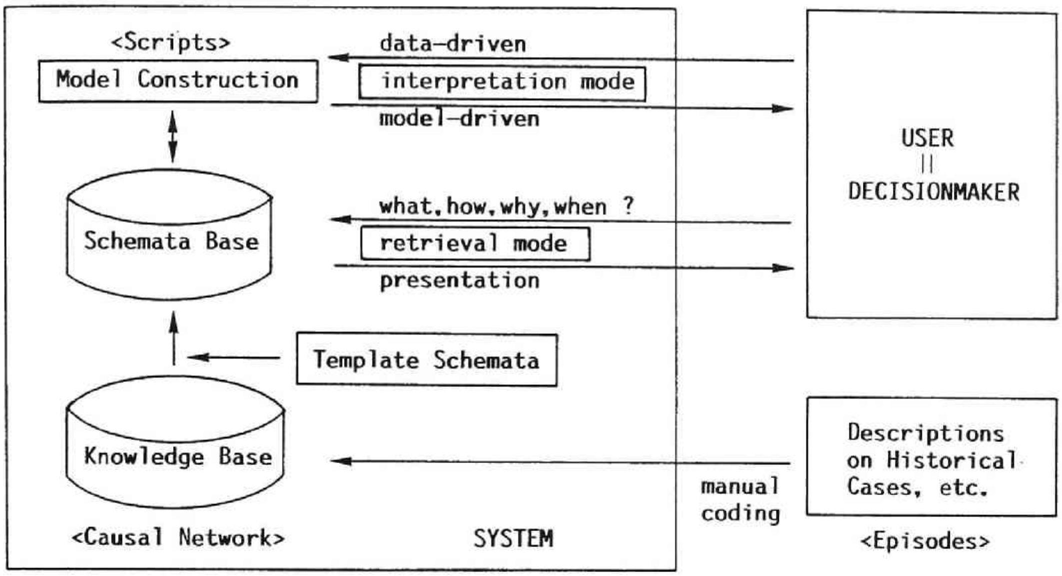


Fig. 9.1 Overview of the system

narrative form, and functions as a model in shaping the world reality. The system can adaptively organize scripts in an inductive manner based on the familiar episodes that are relevant to the input information from the user, i.e., the decisionmaker, and that are stored in the knowledge base. Then, predictions of high-consequence events as well as interpretations of prior episodes are performed based on the constructed script in a hypothses-driven or a model-driven way. Fig.9.1 shows an overview of the system, a lower part of which has been described in the previous chapters (cf. Sawaragi, Iwai and Katai, 1987a; Sawaragi and Iwai, 1987a, 1987b).

9.2 Heuristics in Human Decisionmaking

9.2.1 Cognitive Process in Human Decisionmaking

Simon's theory of "bounded rationality" asserts that cognitive limitations force decisionmakers to construct simplified models in order to deal with the world (Simon, 1977). People behave rationally with respect to this simplified model, and such behaviour is not even approximately optimal with respect to the real world. The model addresses the problem at the abstract, or "information-processing" level, and is a particular formal structure for representing information.

The experimental results indicate that people systematically violate the principles of rational decisionmaking when judging probabilities, making predictions. Errors of predictions are:

1. People seem to rely almost exclusively on specific information and neglect prior probabilities.

2. The variance of one's predictions should be sensitive to the validity of the information on which the predictions are based.

3. People have greater confidence in predictions based on highly redundant or correlated predictor variables.

These violations can be traced to the use of above-mentioned model-driven strategies that human decisionmaker takes to overcome the limitations of their information processing capability. In particular, the active processing of information occurs in a memory of limited capacity, duration. As a result, people appear to utilize heuristics that serve to keep the information-processing demands of a task within the bounds of their limited cognitive capacity. The main three of these heuristics are described below.

(1) Simplifying Heuristics

The effects of increasing the amount of information are to increase the variability of the responces and to decrease the quality of the choices, while also increasing the confidence of the decisionmaker in his judgments.

However, beyond just saying there may be a decrement in performance with increasing information overload, the important psychological question exists concerning how individuals go about dealing with an increasing complex environment with a limited cognitive capacity. It was found that as the task became more complex, the decisionmakers often resorted to simplifying choice heuristics in an effort to reduce cognitive strain. They are essentially trial-and-error learners, who ignore uncertainty and rely predominatly on habit or simple deterministic rules.

(2) Representativeness Heuristics

Kahneman and Tversky have demonstrated that humans often resort to heuristic procedures when facing with such tasks as probability estimation and prediction. In particular, a heuristic called *representativeness* has been shown to be important in the process of intuitive prediction (Kahneman and Tversky, 1972). The idea is that people are likely to think of the outcome that appears most representative of the evidence as the most probable in his prediction. That is, the important determinant seemed to be the degree to which the descriptions were representative of the stereotypes with relatively little regard for prior probabilities or base rate. Kahneman and Tversky regarded this as one of the most significant departures of intuition from the normative theory of prediction.

(3) Availability Heulistics

This heuristic involves judging the probability or frequency of an event by the ease with which relevant instances are imagined or by the number of such instances that are readily retrieved from memory. In life, instances of frequent events are typically easier to recall than instances of less frequent events and likely occurences are usually easier to imagine than unkilely ones.

According to the availability heuristic, one judges "the frequency of a class or the probability of an event by the ease with which instances or occurrences can be brought to mind." Clearly the judgment of how frequently two variables cooccur can be based on the strength of the associative memory bond between them.

The above-mentioned heuristics are closely related to the decisionmaker's knowledge structure as well as to his mental tasks with respect to this structure. That is, the experience for decisionmakers consists of more or less systematic knowledge about it generally as well as the separate experience reported or directly observed with each unique individual on whom decisions must be made. In this stage, human mental tasks called *causal attribution* and *parole decision process* are called into. The attribution theory is concerned with how people interpret information about the causes of events, and with how these judgments are used as well as how these affect the person's behavior (Kelley, 1972). Attribution theory attempts to specify not the real cause of behavior, but how a person infers or attributes cause and what happens once he does. The parole decision process utilizes elements of information provided in the case, which are sought and interpreted by an active process within the decisionmaker (Carroll and Payne, 1976). This process calls on adaptive reorganization of the previous knowledge, beliefs, attitudes, rules of inference, etc. and helps him to direct his examination of the information in a way that confirms to the task structure.

9.3.2 Script Theory as a Behavioristic Decision Model

Note that although such heuristics often lead to correct judgments in an efficient way, it can lead to large and consistent biases that are quite difficult to eliminate. Those heuristics are closely related with the decisionmaker's systematic knowledge structures held inside of them. In consequense, as for a construction of an intelligent DSS, how such biases are produced should be clarified in its relations with the knowledge structure, and how to assimilate such a cognitive process based on the knowledge store will be key issues. If these problems were solved, such an intelligent DSS would be realized that can eliminate undesirable decisionmakers' biases and, at the same time, can use desirable heuristics to reduce their information overload towards satisfactory decisions in a user-friendly manner. For this purpose, the actual behavior of human decisionmaking should be analysed and modelled appropriately. The methodology for this might be based on a descriptive or behavioral decision theory rather than a prescriptive or normative one.

Here, we introduce a script theory as a methodology for modelling human decision behavior, an outline of which was described in chapter two in this thesis. That is, people view decisions based on shallow but nice-sounding rationales as better than decisions based on complex decision-analytic techniques. In other words, the person is not making a simultaneous choice among a set of possible explanations. Instead, he makes a series of choices among sets of promising components that are gradually built into a general, consistent, coherent *pattern* or *scenario*. We may speculate that some specific concrete information has greater power to generate inferences because of the likelihood of such information's calling up "scripts" or schemata involving similar information. The inference then proceeds along the well-worn lines of the previously existing script. Abstract information is probably less rich in potential connections to the associative network by which scripts can be reached. Forecasts and predictions of high-consequence events are often developed with the aid of scenarios. A scenario consists of a series of events linked together in a narrative form.

Scripts are such generalized patternized scenarios which are internalized world model for the decisionmaker adaptively constructed on the basis of individual episodes or experiences. These models, i.e., scripts, are by no means numerical ones that are implemented with esoteric formal techniques but a kind of category formed by a set of individual instances in an inductive manner. Consequently, a model-driven prediction cannot be performed by such a technique as numerical interpolation nor exterpolation but can be formalized using an idea of prototypicality introduced in the previous chapter.

Actually, a prominent characteristic of information-processing psychology has been its reluctance to lose information about individual differences by averaging data over subjects. This attention to the behavior of individual subjects, and consequently to individual differences, is encouraged and reinforced by the methodologies particularly the use of verbal protocols as data and the formalization of theory in computer programs that permit the behavior of individual subjects to be simulated. We believe the origin of researches on artificial intelligence and DSS including expert systems lies in this point. Formal languages for describing the organization of processes and of semantic memory have been developed, that is, the so-called list-processing languages for computers such as LISP and PROLOG have now been used widely and successfully for this purpose.

In the following sections, in order to improve the DSS constructed so far towards a more intelligent one, we will try to develop an methodology for adaptive formulation of scripts from episodic scenarios, which are stored in the schemata base and are organized above the contents of the knowledge base.

9.3 Adaptive Script Learning from Episodes

9.3.1 Concept Formations from Event Schemata

Based on the methodology introduced in Cahapters 6 and 7, an arbitrary episode can be expressed as series of event schemata, which can be transformed into conjunctions of relational statements in the form of [Descriptor=Value]. For instance, the episodic scenario of the example in Chapter 7 can be expressed as

```
EPISODE1:
 contains(SCENARIO1,[EVENT1,EVENT2,EVENT3]),

        EVENT1:<[event-type=revenge],
                    [main-actor=USS],
                    [sub-actor=JSI],
                    [main-content=IncreaseSteelImportsfromJapan],
                    [sub-content=RaiseSteelImportPrices>,
        EVENT2:<[event-type=enablement]
                    [actor=LeagueofSteelCongressmen],
                    [content=SettingTriggerPrices]>,
        EVENT3:<[event-type=successful_enforcement],
                    [main-actor=LeagueofSteelCongressmen],
                    [sub-actor=JSI],
                    [main-content=SettingTriggerPrices],
                    [sub-content=VoluntaryRestrictSteelExports]>,
```

where the first predicate denotes that the episodic scenario consists of

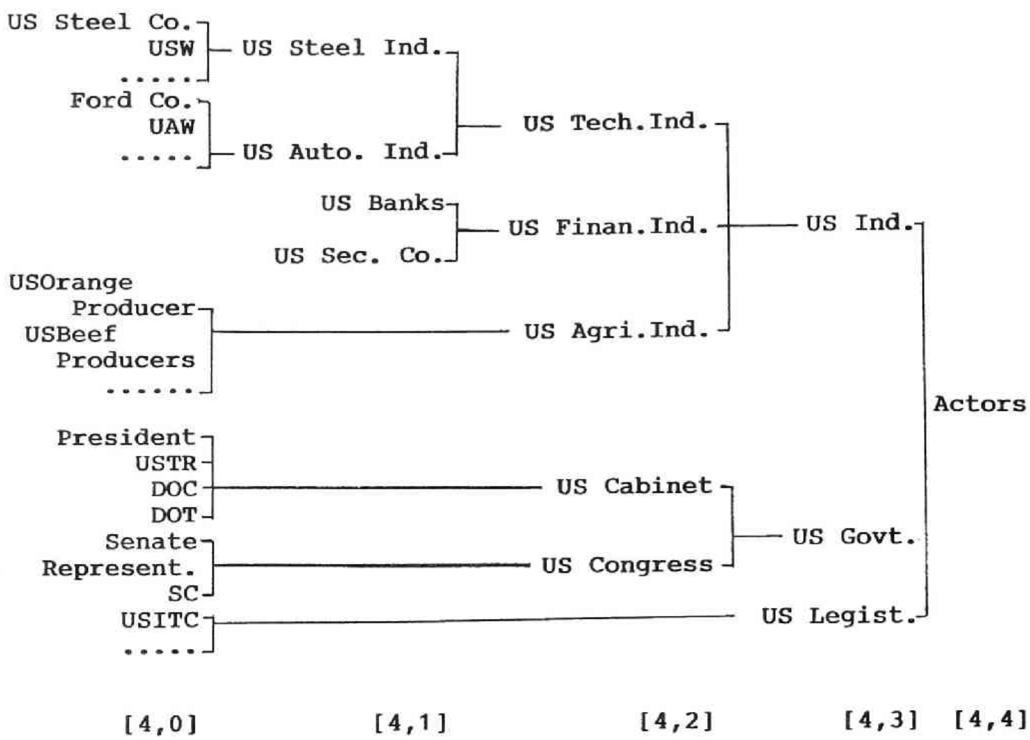


Fig. 9.2 A-kind-of hierarchy for descriptor "actor"

the events in that order and the part enclosed by brackets describes the details of its component event schemata[1]. The value set of such descriptors has a tree-oriented graph structure that reflects the generalization relation between the values, that is, is an *a-kind-of hierarchy*. In our system, the a-kind-of hierarchies for three kinds of descriptors, "event-type," "actor" and "contents" are prepared individually. Fig.9.2 illustrates an a-kind-of hierarchy for descriptor "actor," where the attached numerical tuples represent relative depths of the value in the hierarchy.

From the collections of event schemata composing different episodes, such general descriptions called *generalized event schemata (GES)* can be generated that implies some of event schemata as its instances. We refer to those event schemata implied by GES as *instance event schemata (IES)* hereafter. The generalization proceeds according to the *climbing generalization tree rule* as discussed in the previous chapter. That is,

<CTX & [L = a]>
<CTX & [L = b]>
.....
<CTX & [L = i]>

|<(be generalized to)

<CTX & [L = s]> (GES),

where CTX stands for some arbitrary set of relational statements that are commonly shared by pairs of IES, and s represents the lowest parent node whose descendents include nodes a, b, and i, in the a-kind-of hierarchy for descriptor L. The process is illustrated in Fig.9.3. Note that what can be replaced is not necessarily a unique relational statement as above, but may be a conjunction of some.

By repetitive application of this rule to all the IES as well as to the generated GES, the GES at different levels of generalization are formed and organize a hierarchical tree structure for event concepts where the upper nodes represent such GES whose component descriptors have the more general values in a-kind-of hierarchies and imply the more IES. Fig.9.4(b) illustrates the organized GES above the seven IES shown in Fig.9.4(c). Here, only such GES that imply more than three IES are

[1] As discussed in Chapter 7, the numbers of actors (contents) each event schema specifies depends on its event-types. In the above expression, as for event schemata having two actors (contents), those attributes are expressed in dual, that is, main- and sub-actor (content) to discriminate between them.

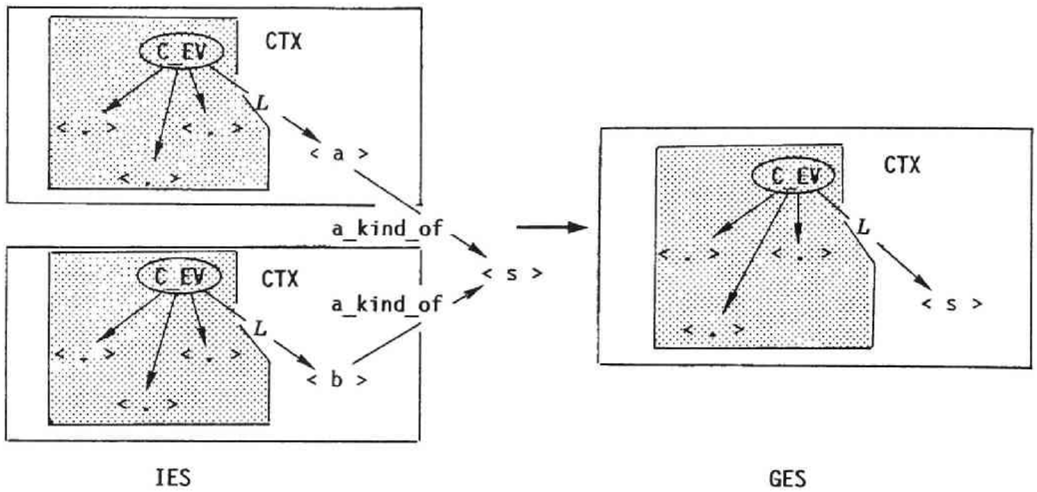


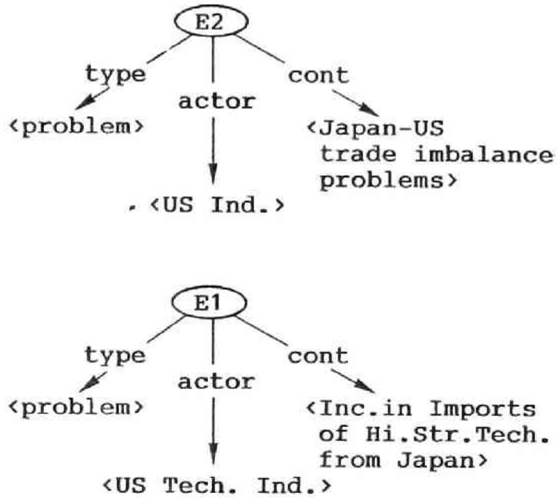
Fig. 9.3 Climbing-generalization-tree rule

marked, E_1 and E_2, both of which are shown in details in Fig.9.4(a). As descrived in the previous chapter, the generalization is performed under the constraint to avoid over-generalization (GP = 5/7, cf. 8.3.2 in the previous chapter). So the organized hierarchical tree structure is determined by the context, that is, the gathered collections of IES to be generalized.

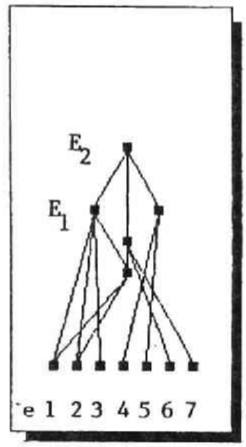
9.3.2 Concept Membership of an IES to a GES

The generated GES represents an event class or an event category that is formed by its instances IES. In general, when describing categories, we cannot treat category membership of each members as a digital, all-or-non phenomenon, but this category is analog, fuzzy one. Moreover, indeed some IES belongs to more than two GES at different

Generalized Event Schemata (GES)

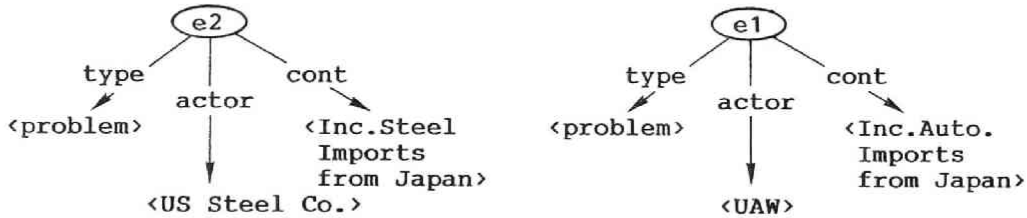


(a)



(b)

Instance Event Schemata (IES)



e_3: [problem,[Motorola,Co.],
[increase,in,communication,devices,from,Japan]]
e_2: [problem,[US,Steel,Co.],
[increase,in,steel,imports,from,Japan]]
e_1: [problem,[UAW],
[increase,in,automobile,imports,from,Japan]]
e_4: [problem,[US,Securities,Companies],
[Japanese,control,of,Euroyen,loan]]
e_5: [problem,[US,Banks],
[Japanese,control,of,Eurobond,market]]
e_6: [problem,[US,Beef,producers],
[closed,beef-market,in,Japan]]
e_7: [problem,[US,Orange,producers],
[closed,orange-market,in,Japan]]

(c)

Fig. 9.4 Generation of GES from IES

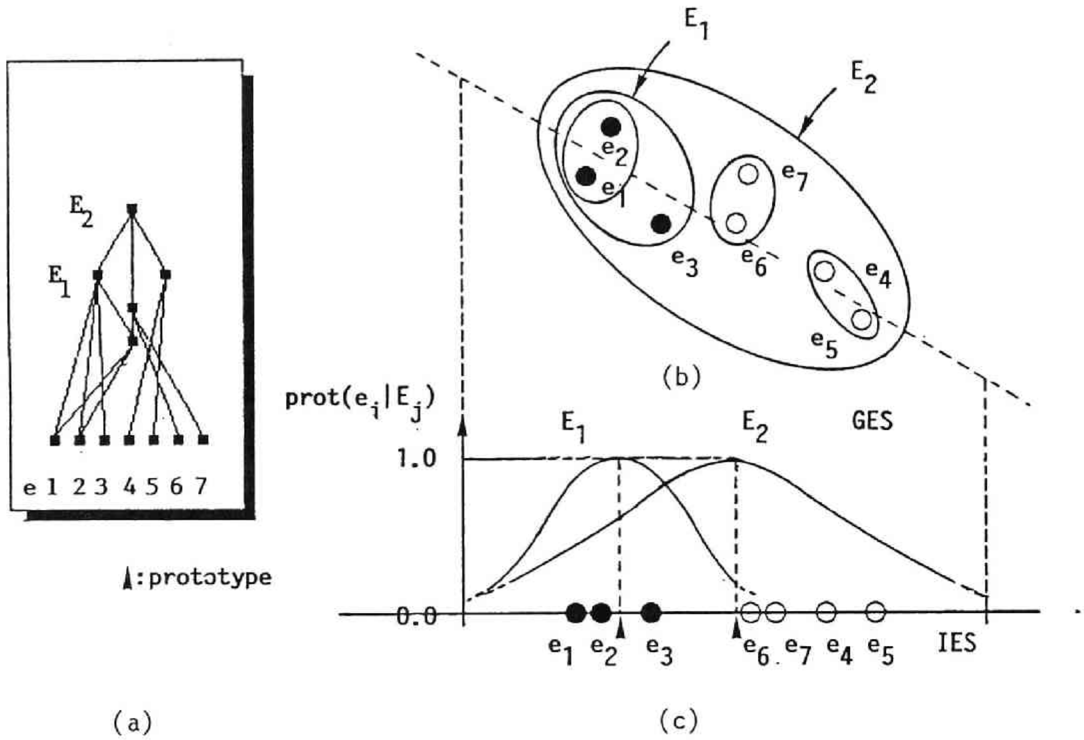


Fig. 9.5 Schematic representation of the organized event classes from instances of Fig.9.4(c)

	e_1	e_2	e_3	e_4	e_5	e_6	e_7	e_t
E_1	0.76	0.76	0.53	0.37	0.37	0.40	0.40	0.76
E_2	0.46	0.46	0.46	0.50	0.50	0.54	0.54	0.46

Table 9.1 Calculated prototypicality of IES of Fig.9.4(c)

generalization levels, but how representative it is in each category varies. For instance, consider an IES, e_1, of Fig.9.4 which forms not only category E_1 with other similar IES e_2 and e_3 but also another category E_2 with more sparsely distributed IES e_3 - e_7. In such a case, it is natural to consider that a representativeness of e_1 is determined not by itself but by other members composing the category. We call this category membership (or representativeness in a category) *prototypicality* (ex., Rosch and Lloyd, 1978). Suppose the schematic representation of the space on which IES of Fig.9.4 are plotted as illustrated in Fig.9.5(b). Being provided with the individual instances as stimuli, such a mental category is formed that more or less implies instances other than provided as shown in the figure. Our psychological category is considered to be formed around this prototype. A prototype is a member that most reflects the redundant structure of the category as a whole having attributes which overlap those of other members of the category. Members of a category come to be viewed as prototypical of the category in proportion to the extent to which they bear a resemblance to this prototype. Therefore, the prototypicality varies not only among individual IES belonging to the same GES, but also among GES to which a specific IES belongs at the same time. This is illustrated in Fig.9.5(c), where IES of Fig.9.5(a) are represented as if they were lying on a dimensional axis.

Based on these ideas, we calculate the prototypicality of IES e_i under GES E_j as follows: for a member IES, e_i, we define a set of relational statements as a factual set of e_i, $FS(e_i)$, which includes not only those composing IES's but also the ones concerning their upper values in the a-kind-of hierarchies. Then, comparing the $FS(e_i)$ for all members of E_j, we can make such a set of common relational statements that are commonly

```
### INPUT INSTANCE TO CALUCULATE ###
    [problem, UAW, IncJaAuExp]

*** ATTRIBUTES SETS FOR [problem, UAW, IncJaAuExp] ***
    [event, problem]
    [event, Event]
    [actor, UAW]
    [actor, USAutoInd]
    [actor, USStrInd]
    [actor, USTechInd]
    [actor, USIndustries]
    [actor, Actor]
    [content, IncJaAuExp]
    [content, IncJaStrProExp]
    [content, IncJaHiStrTechExpUS]
    [content, TrdFriPrbJaUS]
    [content, Content]

### UNDER WHICH SCHEMA ? ###
    [problem, USTechInd, IncJaHiStrTechExpUS]

*** MAJOR ATTRIBUTES OF [problem, USTechInd, IncJaHiStrTechExpUS] ***
    [event, problem]
    [event, Event]
    [actor, USStrInd]
    [actor, USTechInd]
    [actor, USIndustries]
    [actor, Actor]
    [content, IncJaStrProExp]
    [content, IncJaHiStrTechExpUS]
    [content, TrdFriPrbJaUS]
    [content, Content]

*** PROPORTION OF #INTERSECTION/#UNION ***        10/13

### TYPICALITY OF [problem, UAW, IncJaAuExp]
   UNDER [problem, USTechInd, IncJaHiStrTechExpUS] ###        0.76
```

Fig. 9.6 Factual set for IES e_1 and prototype description for GES E_1

shared by no less than half of the members, and we define it as a prototype description of E_j, $PD(E_j)$. $prot(e_i|E_j)$ can be defined as a degree of semantic similarity of e_i to the prototype according to the following:

$$prot(e_i|E_j) = \frac{(PD(E_j) \cap FS(e_i))^{\#}}{(PD(E_j) \cup FS(e_i))^{\#}} \qquad (9\text{-}1)$$

$$(0 \leqq prot(e_i|E_j) \leqq 1.0),$$

where $\cap$ and $\cup$ denote an intersection and an union in a set-theoretical sense, respectively, and the superscript # means the number of elements of the set. Fig.9.6 illustrates a FS for IES e_1, and a PD for GES E_1, respectively. The calculated prototypicality for IES of Fig.9.4 and newly provided IES e_t:

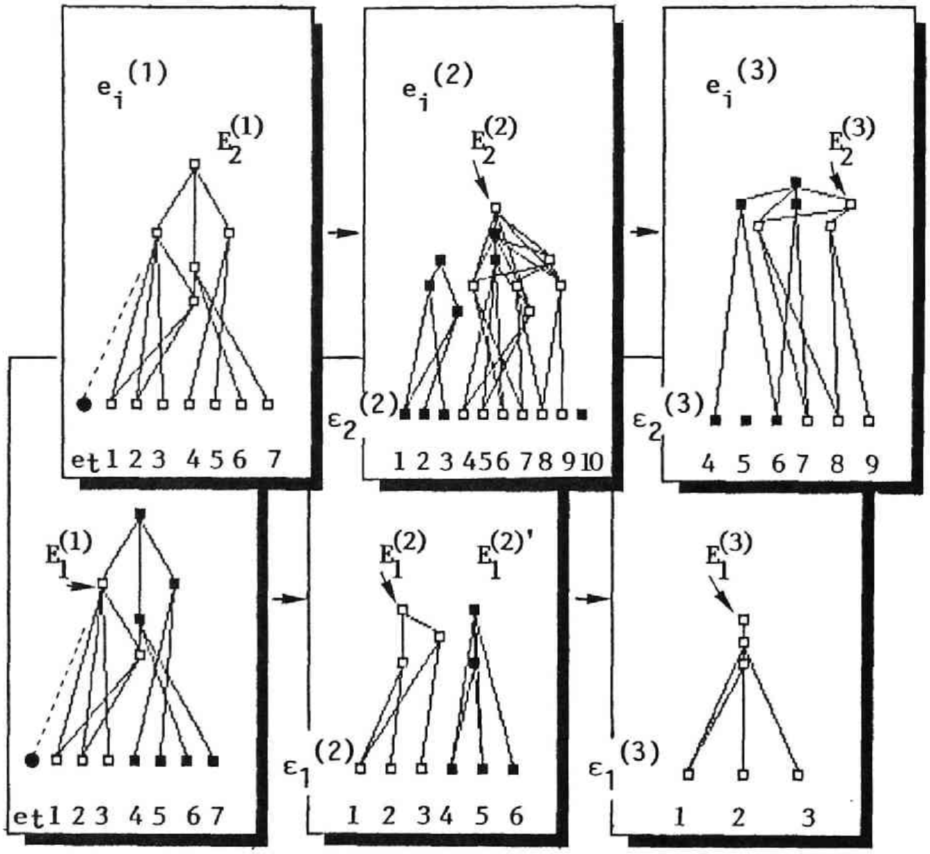
<[event-type=problem],
[actor=UCW(UnionofCommunicationWorkers)],
[content=IncImportsElectricDevicesfromJapan]>,

which is not provided when a category formation is performed, are shown in Table 9.1.

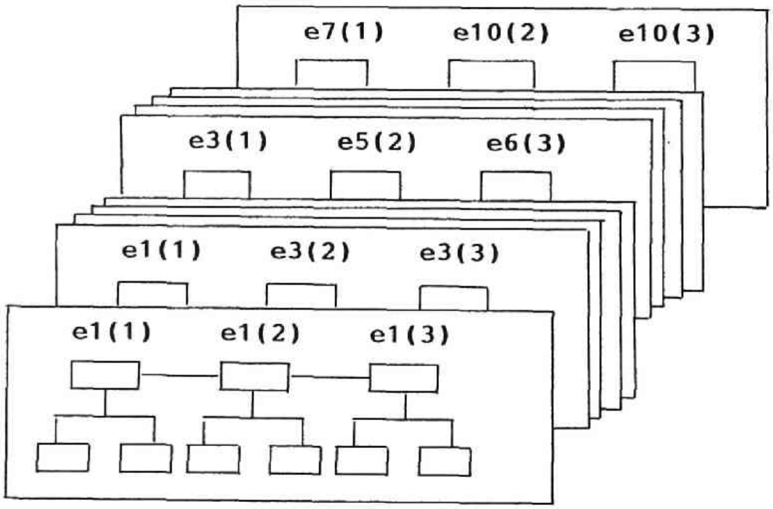
The above procedure of the calculation is almost the same as the process taken in the example of conceptual clustering in the previous chapter, where FS and PD are collections of component schemata of bodies organized through the generalization process and global descriptors for the cell.

9.3.3 Formation of Generally Expected Scenarios

When some newly observed event is given, the consequent expected scenarios that are predictable coherent sequences of generic events can be generated by successive applications of the above mentioned generalization process. We call those scenarios *scripts*. Suppose that for a provided e_t,

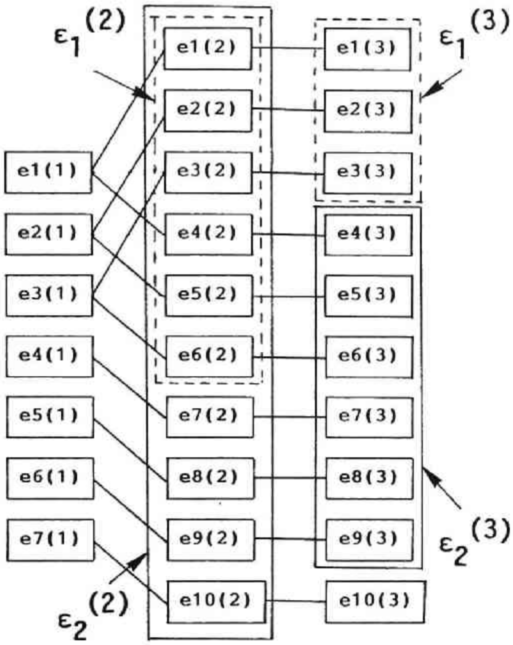


(a)



(b) (continue to next page)

Fig. 9.7 Script formation process

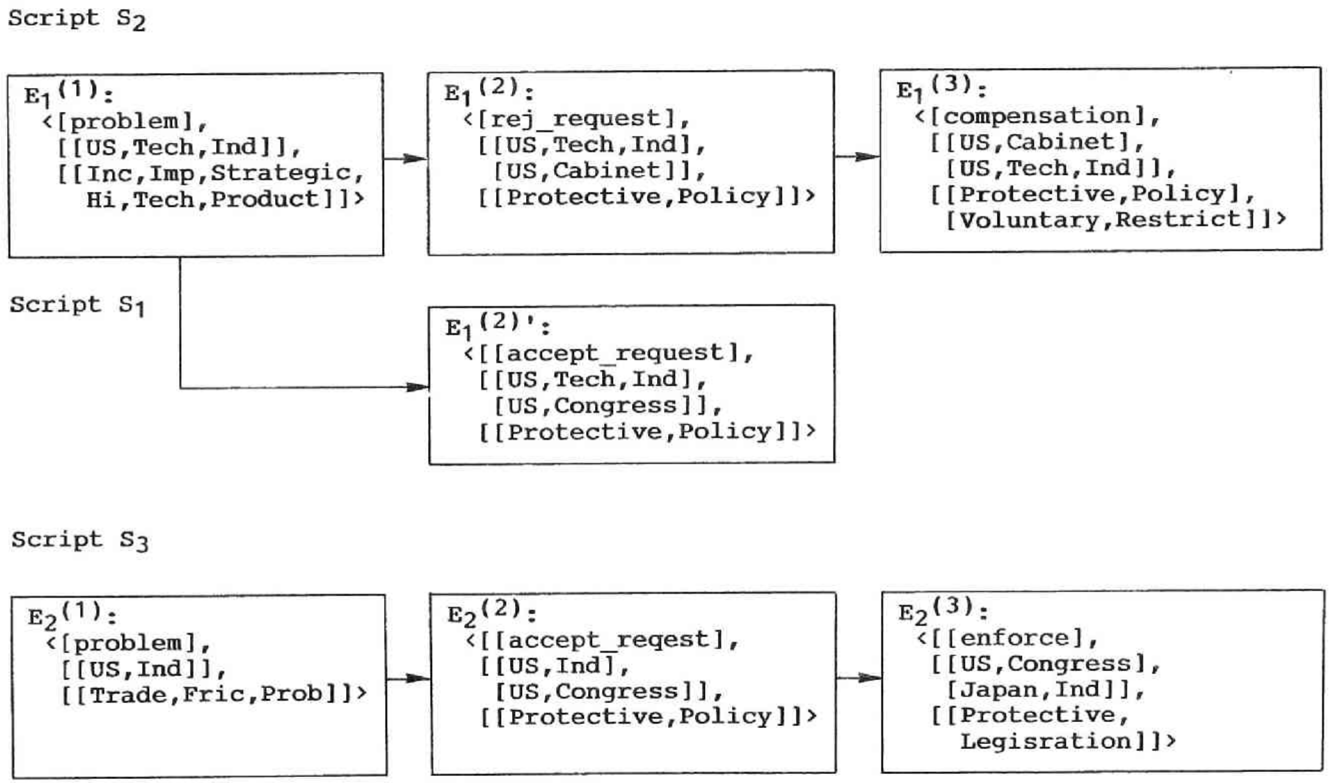


a collection of episodes

(c)

which we call "trigger event" hereafter, ten episodic scenarios in Fig.9.7(b) are retrieved from the schemata base, all of which are preceded by the IES semantically similar to e_t and are now assumed to be the same with the IES of Fig.9.4. Fig.9.7(c) summarizes the causal connectivities of Fig.9.7(b) in digraphs, where IES denoted by $e_i^{(j)}$ is the jth IES appearing in the episodic scenario. Fig.9,7(a) illustrates the procedure for adaptive formation of scripts. From the stored IES $e_1^{(1)}$ - $e_7^{(1)}$ as well as from e_t, GES $E_k^{(1)}$ (k=1,...,n) are organized at various generalization levels. At first, for the members of $E_1^{(1)}$, that is, $e_1^{(1)}$ - $e_3^{(1)}$, another set of IES, $\varepsilon_1^{(2)}$, is determined whose elements are those immediately caused by one of the IES implied by $E_1^{(1)}$ in a collection of retrieved episodic scenarios. In this example, $\varepsilon_1^{(2)} = [e_1^{(2)}, e_2^{(2)}, e_3^{(2)}, e_4^{(2)}, e_5^{(2)}, e_6^{(2)}]$. Then, to this set $\varepsilon_1^{(2)}$, the generalization process is applied. If such GES $E_1^{(2)}$ is generated that can imply no less than half of the IES of $\varepsilon_1^{(2)}$, an event class described by GES $E_1^{(2)}$ can be regarded as the one uniquely predictable when an event belonging to $E_1^{(1)}$ happens, and form a part of script, $E_1^{(1)} \rightarrow E_1^{(2)}$. Note that when more than one GES can be formed as an expected event class as $E_1^{(2)}$ and $E_1^{(2)'}$ in the figure, the script is branched as $E_1^{(1)} \rightarrow E_1^{(2)}$ and $E_1^{(2)'}$. In the same way, a set of IES, $\varepsilon_1^{(3)}$, is determined for the members of $E_1^{(2)}$, and the generalization process is applied to this $\varepsilon_1^{(3)}$. But, it may frequently occur that IES of $\varepsilon_1^{(1)}$ are not so similar with each other and GES implying almost observed IES of $\varepsilon_1^{(1)}$ cannot be formed, which means that an event class cannot be expected further from $E_1^{(1)}$. When this happens, the script preceded by $E_1^{(1)}$ terminates, and another upper GES $E_k^{(1)}$ (k=2,3,..) is tried whether any script can be formed preceded by it at this k-th level. In this way, at each level of GES formed above a given IES e_t, the scripts S_k are organized in parallel as far as the predictable event class can be organized at each level. That is, if the system adopt the lower GES $E_k^{(1)}$, a script from the narrower viewpoint is generated on the basis of the fewer episodic scenarios, while a more global script is formed from more episodic scenarios when it adopts the upper GES $E_j^{(1)}$ (j>k). Fig.9.7(d) shows the generated scripts from the ten episodic scenarios of Fig.9.7(a), which are retrieved from the schemata base when provided with a trigger event e_t, an event illustrated in Table 9.1. Note that although $E_1^{(1)}$ is a subset of $E_1^{(2)}$, the sets of episodes based on which scripts are formed would not necessarily maintain the set-inclusion relationship. Therefore, the scripts following them might differ from each other; in the above example, a "liberalistic" script S_2 ($E_1^{(1)} \rightarrow E_1^{(2)} \rightarrow E_1^{(3)}$) and a "protective" one S_3 ($E_2^{(1)} \rightarrow E_2^{(2)} \rightarrow E_2^{(3)}$) are generated according to whether e_t is interpreted as a member of $E_1^{(1)}$ or $E_2^{(1)}$, respectively.

(continued)


Script S_2
$E_1(1)$:
<[problem],
[[US,Tech,Ind]],
[[Inc,Imp,Strategic,
Hi,Tech,Product]]>
$E_1(2)$:
<[rej_request],
[[US,Tech,Ind],
[US,Cabinet]],
[[Protective,Policy]]>
$E_1(3)$:
<[compensation],
[[US,Cabinet],
[US,Tech,Ind]],
[[Protective,Policy],
[Voluntary,Restrict]]>
Script S_1
$E_1(2)'$:
<[[accept_request],
[[US,Tech,Ind],
[US,Congress]],
[[Protective,Policy]]>
Script S_3
$E_2(1)$:
<[problem],
[[US,Ind]],
[[Trade,Fric,Prob]]>
$E_2(2)$:
<[[accept_reqest],
[[US,Ind],
[US,Congress]],
[[Protective,Policy]]>
$E_2(3)$:
<[[enforce],
[[US,Congress],
[Japan,Ind]],
[[Protective,
Legisration]]>

(d)

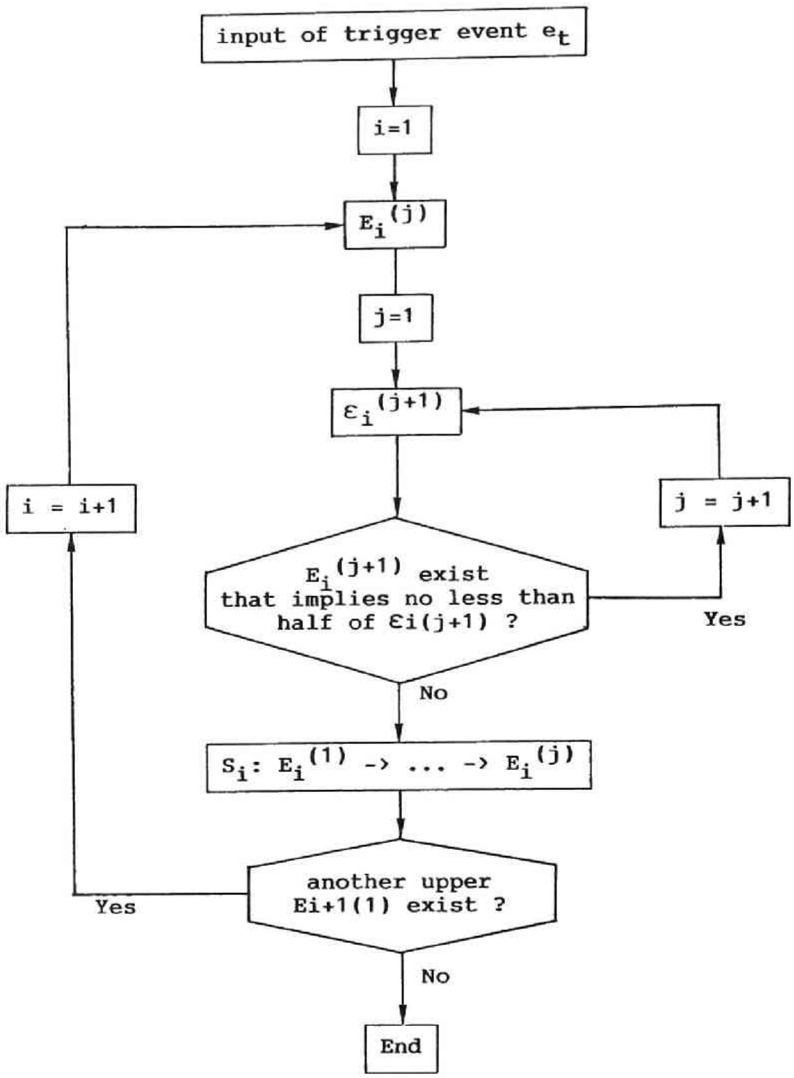


Fig. 9.8 Algorithm for script formation

Fig.9.8 shows an algorithm for the above mentioned adaptive formation of scripts.

9.3.4 Background Ideas of Script Formation

The generated script is actually a category as a set of episodic scenarios. However, different from the categorization of instances dealt with in the previous chapter, this script formation is a categorization of instances with causal order and can be regarded as an automatical extraction of causal attribution. That is, each event is described by a variety of attributes. At first, in the event schemata generalization process, the correlations among attributes are derived, i.e., which attribute values cooccur with which ones. Then, by the successive adaptive selection of IES and the application of the generalization to those IES, causal attribution between two events is made clear that explains which attribute values in the antecedent determine the occurrence of some attribute values in the consequent event.

Such a categorization process may indeed fail to capture an rarely-happening, exceptional scenario. However, as far as there exist some scenarios similar with each other, they are to be aggregated into a corresponding script even though their absolute frequencies are relatively low in a whole repertoire of episodic scenarios.

Moreover, if the retrieved scenarios are those that have no consistency among them on the causal connectivities, they fail to be aggregated into any scripts. But, the mechanism of human cognition is also the same for such noisy experiences without any regularity. Especially in facing an ill-defined complex social reality, it is practically required to eliminate such noisy information and guide the decisionmaker for their behavior or let him focus his attention according to the general patternized trends.

In summary, what plays a great role in supporting decisions in ill-defined problems is by no means a frequency or distribution measured on the basis of the differences among events in their surface meaning, but an investigation into correlations or causal attributions based on its deep, conceptual meaning. This is the reason why the so-called knowledge information processing approach should be taken as a methodology for an intelligent DSS.

9.4 Decision Support System Based on Script Theory

9.4.1 Cognitive Modelling of Human Decisionmaking

The generated scripts are internalized world model that human decisionmaker constructs when he encounters some specific situation of e_t. Each model maps further coming information, which may be either any stored prior episodes or hypothetical one that are not found in the schemata base, onto their corresponding interpretations. Actually, human decisionmakers, being triggered by the occurrence of some event, expect a definite number of scenarios that are likely to follow it, and predict and/or interpret the future events occurrence, or plausibility of prior events' occurrence, in terms of those expected scenarios. Therefore, we can assume that some kind of "tuning function" is constructed for each script according to episodic scenarios. The tuning function for a globally-viewed script is broad, while the one for a narrowly-viewed script is more finely tuned relative to the former as shown in Fig.9.9, where the three scripts, S_1 - S_3, correspond to the ones of Fig.9.7. For instance, while a hypothetical episode H_1 reveals a less strong activation score for the global script S_3 than another episode H_2, the former will reveal the higher score for the specific scripts, S_1 and S_2 than the global one will.

As for a simulation system of such a cognitive mechanism of human decisionmaking, the most promising is a system in which any hypothetical episodes can be mapped onto activation scores for the appropriate script. For this purpose, we have to formulate the following two issues: how the

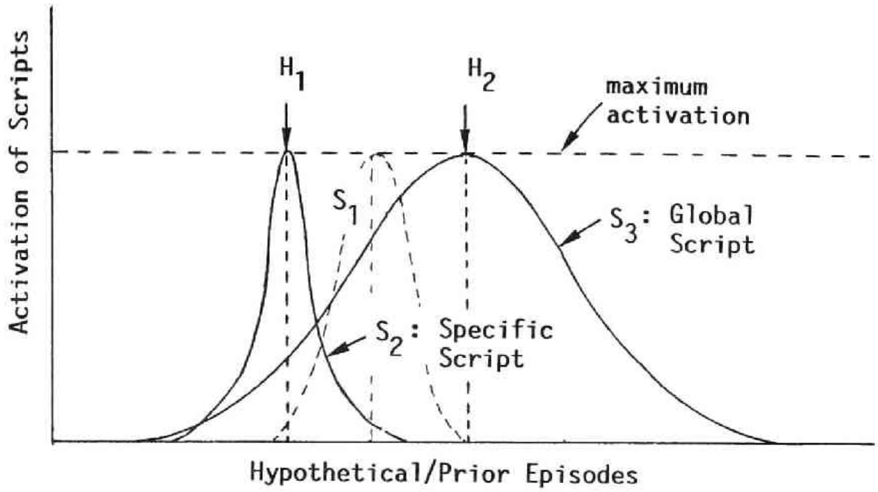


Fig. 9.9 Tuning functions for generated scripts of Fig.9.7

appropriate script is selected among the parallely-activated scripts, and how fitness of episodes to that script is calculated. With respect to those two issues, we suggest the procedures that are based on the heuristics, availability and representativity. and their related mental tasks mentioned in 9.2.1 in the following section.

9.4.2 Script Selection Process

We call this script selection process a *satisfactory procedure*, which requires two valued characteristics of a script. These are the relative accessibilities of alternative scripts and the magnitude of the fitness of the current case to the script. The former is a degree of easiness to be reminded of, i.e., a measure for availability, and is determined by the nature of the script itself, while the latter depends on the type of the case and is a measure for representativity.

We assign relative accessibilities to a set of generated scripts as a function of the tradeoff between two competing characteristics, the generalization level at which the script is generated and the number of episodic scenarios in the schemata base from which the script is generated. That is, we assume that the more concrete the scenario is and the more instances are observed in the past, in a word, the more redundant structure it is, the more easily the script can be reminded of, in other words, the higher availability it reveals.

Setting the relative accessibilities of scripts determines the order in which the coming information is checked for its fit to the scripts. Let us denote the coming information, i.e., the hypothetical episode, to be predicted and/or interpreted as a set of conjunctive IES, $e_{p1},..,e_{pk}$ $(k=1,2,..)$. That is, when the triggering event e_t happens, the occurrence of some specific event $(k=1)$ or the cooccurrence of more than two events $(k \geqslant 2)$ are to be predicted.

The fitness of a hypothetical scenario H, i.e., how much representative H is as an expected scenario when e_t happens, can be calculated using $prot(e_{pi}|E_j)$ of Eq.(9-1) according to the following:

$$prot(H|S_i) =$$

$$prot(e_t, e_{p1},..,e_{pk}|S_i) = \frac{1}{k+1}[prot(e_t|E_1) + \sum_{i=1}^{k}[\max_j prot(e_{pi}|E_j)]], \qquad (9\text{-}2)$$

where E_j denotes j-th GES composing script S_i.

Normally, the probability of a multievent scenario's happening is a multiplicative function of the probabilities of the individual links. The more links there are in the scenario, the lower the probability of the entire scenario's occurrence. The probability of the weakest link sets an upper limit on the probability of the entire narrative.

Human judges do not appear to evaluate scenarios according to these normative rules. It is reported that the probability of a multilink scenario is judged on the basis of the average likelihood of all its links (Carroll and Payne, 1976). Subsequent strong links appear to "even out" or compensate for earlier weak links, making it possible to construct scenarios with perceived probabilities that increase as they become longer, more detailed, and normatively less probable. These results suggest that scenarios which tell a "good story" by burying weak links in masses of coherent detail may be accorded much more credibility than they deserve.

The script to be selected is the first script that has a satisfactory fit, that is, script S_i $prot(H|S_i)$ is no less than the satisfactory threshold k. Once S_i is selected as the interpretation of the hypothetical episode, S_{i+1} is never even checked and therefore cannot be selected even though S_{i+1} provided the best fit. This reflects the efficiency of human information processing, even if it might sacrifice the optimality. Fig.9.10 summarizes the procedure.

Once a case is determined to be interpreted in terms of the selected script, the decisionmaker is able to make predictions about previously unobserved aspects of the case. These predictions can be useful since the partial specification of the case might not have constituted an adequate guide for behaviour, and the prototypicality can help him identify and choose among future courses of actions.

9.4.3 Experientially Constructed Interactive decision Support Based on Script Theory

Based on the cognitive model of the script selection process, we are now developing a prototype of an intelligent decision support system with

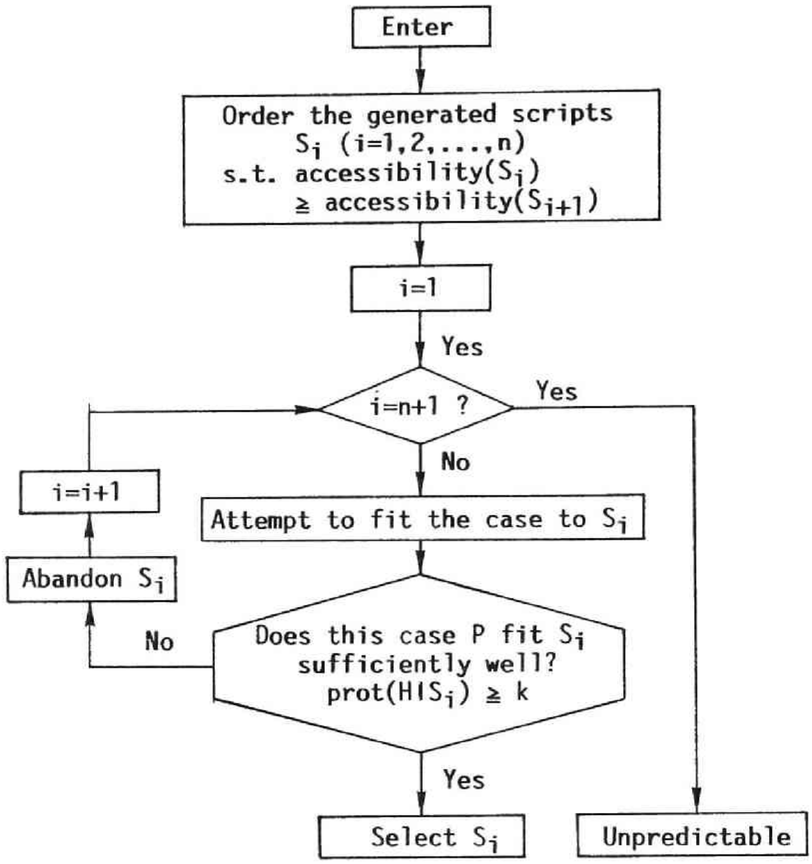


Fig. 9.10 Script selection process

respect to the same problem domain as dealt with in Chapter 7. The system supports the decisionmaker's predecision stages by simulating his cognition based on a vast amount of the stored prior episodes. The conversation process between the decisionmaker and the system is summarized in Fig.9.11.

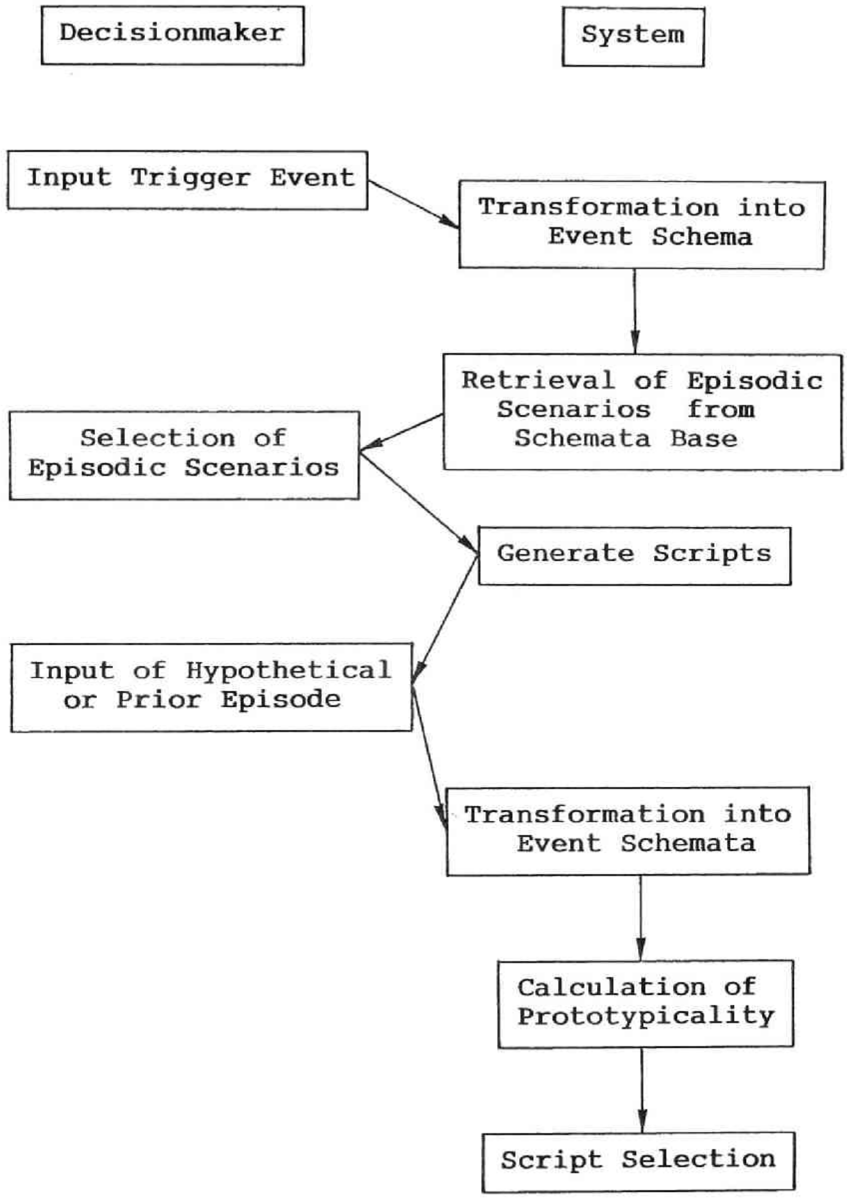


Fig. 9.11 Interactive conversation process of the system

At first, the decisionmaker inputs a triggering event, whose consequent scenario he wants to predict, in natural language expressions. This sentence is parsed and transformed into an appropriate event schema according to the same procedure described in Chapter 7. Having identified the trigger event, the system searches the episodic scenarios in the schemata base that are preceded by the events similar to the triggering event , and presents them to the decisionmaker in sentential expressions. Then, the system asks the decisionmaker whether he accepts all the retrieved scenarios or selects some out of them. In case he accepts all, the system is to generate scripts based on all the retrieved scenarios, so it is useful when he wants to ask the system to generate all possible interpretation based on all available prior data. However, it is often the case that the desirability of each retrieved scenarios for the decisionmaker varies. Since the decisionmaker has his own personal values or utilities, he can judge which scenarios in the past are good and which are not. In consequence, he might persue the interpretation from his own viewpoint. In this case, the decisionmaker selects some of the retrieved scenarios based on his intuitive criteria. From those selected episodic scenarios, the system tries to generate scripts. Since those scripts are consistent with the decisionmaker's utility. the interpretation driven by those scripts would also reflect his own perspective. Next, the decisionmaker enters the events to be predicted and/or interpreted in natural language expressions, which are transformed into event schemata. Finally, calculating their prototypicality under each scripts, the system selects the satisfactory one. Under this selected context, the system performs a number of intellectual tasks such as further prediction of the consequence of his assuming events or automatic planning of scenarios, which are now being implemented.

Fig.9.12 and Fig.9.13 illustrate the conversation process with respect to the same example of Fig.9.7. Each has windows for the presentation of the input triggering event that is transformed into a form of event schemata (Window 1), and of the generated scripts (Window 2) as well as the window where the conversation is displyed (Window 3).

In Fig.9.12(a), the retrieved ten episodic scenarios are all accepted by the decisionmaker and, as a result, two GES, $E_1^{(1)}$ and $E_2^{(1)}$ are organized above the triggering event from which three scripts S_1 - S_3 are formed. These three scripts are ordered in [S_1, S_2, S_3] according to their accessibilities. In reply to the input of information to be predicted, the system tries to fit it to the firstly accessible script S_1, which is a "protectional" script, but since the calculated prototypicality is quite low (0.31), it is rejected and another script S_2 is attempted. In this case, the information matches well with this "liberalistic" script (0.71) and the system predicts it would be possible to happen. Following it in the figure, the coocurrence of two events is predicted, which fits to S_2 (0.76) better than the above case. Fig.9.12(b) illustrates the above procedure with respect to the tuning functions corresponding to the three scripts.

Fig.9.13 illustrates a case where just the protectional episodic scenarios are selected out of the retrieved ones by the decisionmaker. As shown in Fig.9.13(a) and (b), just two scripts are formed both of which are protectional ones. Consequently, for the same input of the hypothetical episode as the one of Fig.9.12, the system fails to match it with any scripts and negates the possibility of the occurrence of the events in this case.

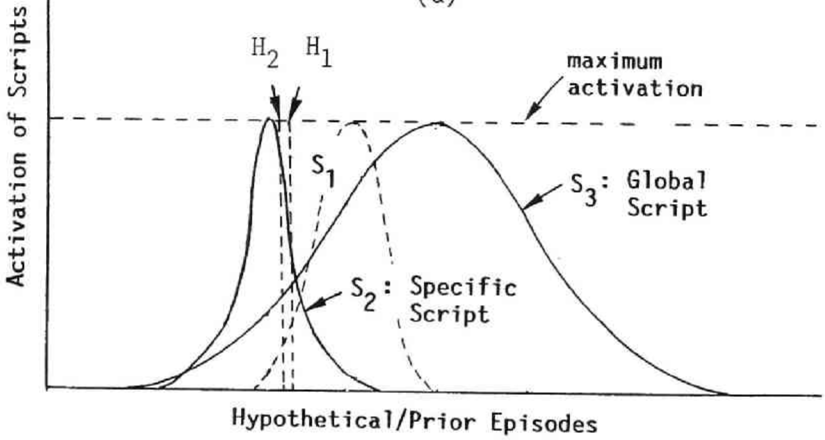
In the conventional decision theories, there did exist the methodologies to extract the decisionmaker's subjective utility or decision

```
# Trigger Event #

problem
 [UCW]
 [IncJaImpElecDevice]
```

```
               |-> E1(2)"
E1(1)  --|
               |-> E1(2)  --> E1(3)

E2(1)   ---> E2(2)  --> E2(3)
```

```
Input trigger event
   |: UCW is suffering from increase in imports of
      electric devices from Japan.

Event is interpreted as event schema:
  [problem,
   [UCW]
   [IncJaImpElecDevice]]
There are 10 episodes preceded by the event.
Will you choose some of them ?
|: no.
Generalized event schemata of Trigger event are:
   E1(1): [problem, [USTechInd], [IncJaImpStrHiTech]]
   E1(2): [problem, [USInd], [TradeFrictionProb]]
3 scripts are generated.
|: Predict whether USPresident rejects dumping suit for IC to UCW. ----- H1
   [rej_request, [UCW, USPresident], [RejDumpSuitIC]]
   Script 1 is tried, prototypicality is 0.31, rejected.
   Script 2 is tried, prototypicality is 0.71, accepted.
--> Definitely possible.

|: Predict whether USPresident rejects dumping suit for IC to UCW, and USPresi
dent compensates UCW for rejectindg dumpig suit for IC by beginning  agreement
 of self_imposed control of IC.                                      ----- H2
   [rej_request, [UCW, USPresident], [RejDumpSuitIC]]
   [compensate, [USPresident, UCW], [BeginAgreeSelfImposeContIC]]
   Script 1 is tried, prototypicality is 0.25, rejected.
   Script 2 is tried, prototypicality is 0.76, accepted.
--> Definitely possible.
```

(a)

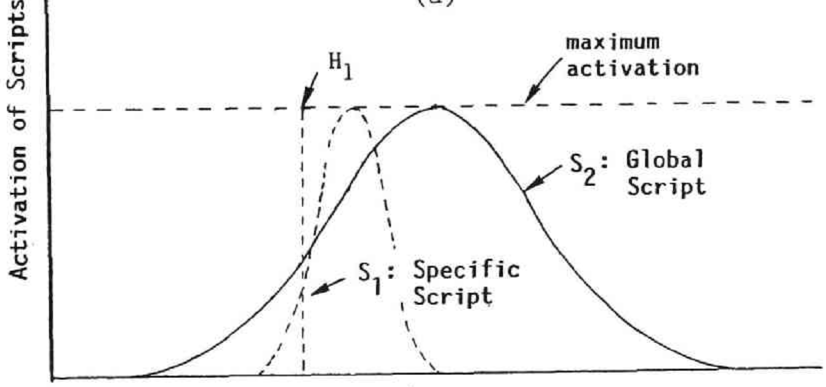


(b)

Fig. 9.12 Illustration of the conversation in case all episodes are selected

```
# Trigger Event #                E1(1)   ---> E1(2)

problem                          E2(1)   ---> E2(2)  --> E2(3)
 [UCW]
 [IncJaImpElecDevice]

Input trigger event
   |: UCW is suffering from increase in imports of
      electric devices from Japan.

Event is interpreted as event schema:
  [problem,
   [UCW]
   [IncJaImpElecDevice]]
There are 10 episodes preceded by the event.
Will you choose some of them ?
|: yes.

Generalized event schemata of Trigger event are:
   E1(1): [problem, [USTechInd], [IncJaImpStrHiTech]]
   E1(2): [problem, [USInd], [TradeFrictionProb]]
2 scripts are generated.
|: Predict whether USPresident rejects dumping suit for IC to UCW. ------ H1
   [rej_request, [UCW, USPresident], [RejDumpSuitIC]]
   Script 1 is tried, prototypicality is 0.31, rejected.
   Script 2 is tried, prototypicality is 0.35, rejected.
--> Definitely impossible.
```

(a)



(b)

Fig. 9.13 Illustration of the conversation in case some episodes are selected attitudes auch as a multi-attributes utility theory. In these theories, the problem with multiple attributes is divided into a set of partial problems with uni-attribute, and the decisionmaker's utility with respect to the whole problem can be calculated by synthesizing those partial ones. The decisionmaker is obliged to answer the system's queries with precise quantified units, but this is a hard task for decisionmaker's intuitive judgment.

In our system, the extraction of the decisionmaker's utility is performed by just a binary decision, i.e., "accept this episodic scenario or not," "good or bad" etc., it can sufficiently stimulate his intuition. And, since it is performed through concrete episodic scenario, the decisionmaker has a realistic image keeping his global views of the problem without division. We believe such a feature is really required in the design of DSS in the future.

9.5 Concluding Remarks

An ill-structured problem is defined as one putting demands on knowledge as prior experiences, and what is considered as evidence of some judgment's truth or plausibility is the empirical knowledge itself rather than any deductive inferences in an analytical way. Moreover, the boundary of required knowledge is not definite, that is, the problem is often "open-ended."

In this chapter, as a methodology to deal with such ill-defined, open world realistic problems, we proposed a general architecture for DSS based on human cognitive models. Human decisionmaker's ways of thinking or problem solving in order to overcome the limitation of information processing capability are really efficient and effective. Of course, it is the origin that may produce undesirable biases, but as far as this mechanism is based on a vast amount empirical knowledge with the close investigation into its deep conceptual meaning, the reasoning performed by that mechanism must be a satisfactory one for the human decisionmakers.

We believe that our suggested methodology to adaptively construct models based on familiar episodes and to quantify the model-driven interpretation using prototype theory works as a basis for constructing much more intellectual and user-friendly decision support for such ill-structured, open world problems.

CHAPTER 10

CONCLUSIONS

As for a general architecture for DSS, we took notice of a knowledge processing level of integrating a huge amount of expertise in advance of the final stage of choice, and proposed a knowledge-based DSS which helps a decisionmaker understand the problem in the information gathering stage towards sound decisionmaking, and then recommends alternative policy sets or a course of action to be adopted reflecting an individual decisionmaker's own attitude or strategy for his problem solving interactively.

In these stages of actual decisionmaking, the greatest issues that can affect the "quality" of decision are such human factors as information overload caused by human cognitive limitations and psychological stress or dissonance decisionmakers feel when he makes decisions. In this thesis, to develop DSS that can eliminate those undesirable factors, we at first discussed a knowledge representation formalism for societal problems as causal networks, i.e., signed digraphs. As for integration methodologies for these causal knowledge as well as for its utilization for decision support, two different approaches were developed; the one based on the surface knowledge structures and the other one on a basis of the deep knowledge structures.

In the former approach, knowledge was treated just as the one describing qualitative structural relationships, promoting or suppressing, among problem elements, that could be obtained from documents by means of documentary coding. The methods to integrate the causal knowledge and to quantify its strength were proposed purely graph-theoretically, and knowledge retrieval system was presented on a basis of those structural properties. Using these, individual decisionmaker's conformity for the alternatives with respect to the cognitive dissonance was measured and a DSS that recommends the policy alternatives accordingly in an interactive manner was constructed.

In the latter approach, we based on an assumption that knowledge is by no means a closed collection of causal relationships extracted from documents, but is a product resulting from intermingling them with the commonsense we universally have. Consequently, not only causal temporal ordering but also conceptual dimensions such as concrete-abstract one and detailed-aggregated one were emphasized, and methodologies to organize hierarchical deep knowledge structures above the surface knowledge were developed. This made possible to explicate the conceptual and sophisticated principles implicitly represented in the surface expressions and to provide the decisionmaker with a variety of interpretations for an identical piece of knowledge. The knowledge organization through such semantic meaning was much alike human memory structures and are closely related with their cognitive processes, especially with respect to concept formation from instances in a bottom-up way as well as to hypotheses-driven prediction in a top-down manner. In the thesis these processes were modelled based on the hierarchical causal knowledge structures and applied to an interactive DSS. Although the system only stores individual instances, by adaptively reorganizing an appropriate part of its knowledge base, it automatically acts as if it had generalized knowledge even for a particular input outside of its knowledge. Such self-organizing capability was really valid in dealing with an open-ended problem domain and in making decisionmakers focus their attention in the uncertain, complicated reality.

Our suggested architectures of DSS in this thesis were just prototypes, and there remain a number of future problems to be solved towards realizing much more practical and intellectual DSS:

First, the proposed methodologies were in principle assumed for a single decisionmaker. However, it is often the case that the system is required to derive some compromised solution or interpretation for a number of competitive decisionmakers whose goals or utilities might be in conflict. For this problem, our system can be easily extendable. The policy validity evaluation function developed in Part II can be constructed above multi-attributes corresponding to individual's goals. The script constructed for individual decisionmakers in Part III also can be used for a planning problem with multiple actors, which requires a precise formulation on reasoning of goal and plan interferences among actors (ex., Wilensky, 1982).

Second, the derivation of causal assertions from documents depends on the recognition of a coder who analyses statements of documents and rewrites them into cognitive maps composing the knowledge base. Therefore, the credibility of quantities for causal effects derived by our suggested methods depends much on the correctness of the coder's problem analysis and the reliability of the coding. In order to apply our methods to more complex large-scaled problems, the construction of greater knowledge base will be essential, which will evoke such difficult problems as the maintenance of objectivity and the improvement of efficiency from manual coding to automatic one. For these problems, we can expect much from the research in the field of natural language processing on the automatic extraction of causal chain from sentences concerning everyday behaviors (Schank, 1972, 1975; Schank and Riesbeck, 1981).

Finally, as for an adaptive reorganization of knowledge structures, we just dealt with it in a static way. However, it is natural that a decisionmaker's memory structure may be dynamically altered with a further coming information or with a new observation. More concretely, how long the once established script remains valid and when the script is rejected or modified with an unexpected input are fundamental problems to be solved. For these, we believe that the researches on dynamic memory in a domain of cognitive science (e.x., Kolodner, 1983, 1984; Lebowitz, 1983; Schank, 1980, 1982) and the ones on Truth Maintenance System (TMS) and Assumption-based TMS in a field of artificial intelligence (Doyle, 1979 ; De Kleer, 1986) will offer some solution for this in the near future.

REFERENCES

(The ones marked with * are papers by the author and his colleagues concerning with this thesis)

Abelson, R.P.(1973), "The Structure of Belief Systems," in Schank, R. and Colby, K.M. (Eds.), *Computer Models of Thought and Language*, Freeman, San Francisco, 287-340.

Abelson, R.P.(1975), "Representing Mundane Reality in Plans," in Bobrow, D.G. and Collins, A. (Eds.), *Representation and Understanding: Studies in Cognitive Science*, Academic Press, New York, 273-309.

Abelson, R.P.(1976), "Script Processing in Attitude Formation and Decision Making," in Carrol, J.S. and Payne, J.W., *Cognition and Social Behavior*, Lawrence Erlbaum Assoc., Hillsdale, NJ., 33-46.

Abelson, R.P.(1981), "Psychological Status of the Script Concept," *American Psychologist*, **36**, 715-729.

Alter, S.L., (1980), "Decision Support Systems: Current Practice and Continuing Challenges," Addison-Wesley.

Anderson, J.R.(1980), "Cognitive Psychology and Its Implications," W.H. Freeman and Company, San Francisco.

Anderson, P. and Thorson, S.(1982), "Artificial Intelligence Based Simulations of Foreign Policy Decision Making," *Behavioral Science*, **27**, 176-193.

Anderson, J.R.(1981), "Concepts, Propositions, and Schemata: What are the Congnitive Units ?" in Flowes, J.H. (Ed.), *Nebraska Symposium on Motivation*, University of Nebraska Press Lincoln/London, 121-162.

Axelrod, R.(1973), "Schema Theory: An Information Processing Model of Perception and Cognition," *American Political Science Review*, **67**, 1248-1266.

Axelrod, R.(1976), "Structure of Decision: the Cognitive Maps of Political

Elites," Princeton Univ. Press, Princeton, NJ.

Bartlett, F.C.(1932), "Remembering: A Study in Experimental and Social Psychology," The Macmillan Co., New York.

Bennett, J.L. (1983), "Building Decision Support Systems," Addison-Wesley.

Berkeley, D. and Humphreys, P.(1982), "Structuring Decision Problems and the 'Bias Heuristic'," *Acta Psychologica*, **50**, 201-252.

Bobrow, D.G. and Norman, D.A.(1975), "Some Principles of Memory Schemata," in Bobrow, D.G. and Collins, A. (Eds.), *Representation and Understanding: Studies in Congnitive Science*, Academic Press, New York, 131-149.

Bobrow, D.G. and Winograd, T.(1977), "An Overview of KRL, a Knowledge Representation Language," *Cognitive Science*, **1**, 3-46.

Bower, G.H., Black, J.B. and Turner, T.J.(1979), "Scripts in Memory for Text," *Cognitive Psychology*, **11**, 177-220.

Burns, J.R. and Winstead W.H.(1982), "Input and Output Redundancy," *IEEE Transactions on Systems, Man, and Cybernetics*, **SMC-12**, 785-793.

Carbonell, J.G. (1980), "Towards a Process Model of Human Personality Traits," *Artificial Intelligence*, **15**, 49-74.

Carbonell, J.G. (1981), "Subjective Understanding Computer Models of Belief Systems," Umi Research Press.

Carroll, J.S. and Payne, J.W.(1976), "The Psychology of the Parole Decision Process: A Joint Application of Attribution Theory and Information Processing Psychology," in Carroll, J.S. and Payne, J.W. (Eds.), *Congnition and Social Behavior*, Lawrence Erlbaum, Hillsdale, NJ., 13-32.

Cartwright, D. and Harrary, F. (1956), "Structural Balance: A Generalization of Heider's Theory," *Psychological Review*, **63**, 277-293.

Cartwright, D. and Harary, F. (1970), "Ambivalence and Indifference in Generalizations of Structural Balance," *Behavioral Science*, **15**, 497-513.

Chandrasekaran, B. (1984), "Deep versus Compiled Knowledge," *Int. J. Man-Machine Studies*, **20**.

Colby, K.M. (1973), "Simulations of Belief Systems," in Schank, R. and Colby, K.M. (Eds.), *Computer Models of Thought and Language*, Freeman, San Francisco, 251-286.

DeKleer, J. (1986), "An Assumption-based TMS," *Artificial Intelligence*, **28**, 127-162.

Dietterich, T.G. and Michalski, R.S.(1985), "Discovering Patterns in Sequences of Events," *Artificial Intelligence*, **25**, 187-232.

Dietterich, T.G. and Michalski, R.S. (1981), "Inductive Learning of Structural Descriptions: Evaluation Criteria and Comparative Review of Selected Methods," *Artificial Intelligence*, **16**, 257-294.

Dietterich, T.G. and Michalski, R.S. (1984), "A Comparative Review of Selected Methods for Learning from Examples," in Michalski, R.S., Carbonell, J.G. and Mitchell, T.M. (Eds.) *Machine Learning*, Tioga Pub. Co., Palo Alto, CA., 41-81.

Doyle, J. (1979), "A Truth Maintenance System," *Artificial Intelligence*, **12**, 231-272.

Dyer, J.S. (1979) and Sarin, R.F., "Measurable Multiattribute Vlue Functions," *Operations Research*, **27**, 810-822.

Eden, C. and Jones, S.(1980), "Publish or Perish ? — Case Study," *Journal of Operational Research Society*, **31**, 131-139.

Einhorn, H.J. and Hogarth, R.M. (1981), "Behavioral Decision Theory: Processes of Judgement and Choice," *Ann. Rev. Psychol.*, **32**, 53-88.

Embrey, D.E. and Reason, J.T. (1986), "The Application of Cognitive Models to the Evaluation and Predicition of Human Reliability," *Proc. of International Topical Meeting on Advances in Human Factors in Nuclear Power Sustems*, Knoxville, USA, 292-301.

Embrey, D.E.(1986), "SHERPA: A Systematic Human Error Reduction and Prediction Approach," *Proc. of International Topical Meeting on Advances in Human Factors in Nuclear Power Systems*, Knoxville, USA, 184-193.

Feigenbaum, E.A. (1977), "The Art of Artificial Intelligence: Themes and Case Studies of Knowledge Engineering," *Proc. of 5th International Joint Conference of Artificial Intelligence*.

Festinger, L. (1957), "A Theory of Cognitive Dissonance," Row-Peterson, Evanston, MA.

Fishburn, P.C. (1970), "Utility theory for Decision Making," Wiley, New York.

Graesser, A.C. and Nakamura, G.V. (1982), "The Impact of a Schema on Comprehension and Memory," *The Psychology of Learning and Motivation*, Vol.16, 59-109.

Grosz, B.J. (1977), "The Representation and Use of Focus in a System for Understanding Dialogs," *Proc. of 5th International Joint Conference on Artificial Intelligence*, 67-164.

Harary, F., Norman, R.Z. and Cartwright, D. (1965), "Structural Models: An Introduction to the Theory of Directed Graphs," Wiley, New York.

Hart, J.A. (1977), "Cognitive Maps of Three Latin American policy Makers," *World Politics*, **30**, 115-140.

Hayes-Roth, F., Waterman, D.A. and Lenat, D.B. (1983), " Building Expert Systems, Addison Wesley, MA.

Heider, F. (1946), "Attitudes and Cognitive Organization," *Journal of*

Psychology, **21**, 107-112.

Hogarth, R.M., Michand, C. and Mery, J.L. (1980), "Decision Behavior in Urban Development: A Methodological Approach and Substantive Considerations," *Acta Psychologica*, **45**, 95-117.

Holland, J.H. and Reitman J.S. (1977), "Cognitive Systems Based on Adptive Algorithams," *SIGART Newsletter*, No.63, 49-65.

Humphreys, P. and Berkeley, D.(1983), "Problem Structuring Calculi and Levels of Knowledge Representation in Decision Making," in Scholz, R.W. (Ed.) *Decision Making Under Uncertainty*, Elsevier Science Publishers, North-Hololland, 121-157.

Kahneman, D. and Tversky, A. (1972), "Subjective Probability: A Judgment of Representativeness," *Cognitive Psychology*, **3**, 430-454.

Kahneman, D. and Tversky, A. (1973), "On the Psychology of Prediction," *Psychological Review*, **80**, 237-251.

Katai, O. and Iwai, S. (1978), "On the Characterization of Balancing Processes of Social Systems and the Derivation of the Minimal Balancing Processes," *IEEE Transactions on System, Man and Cybernetics*, **SMC-8**, 337-348.

* Katai, O., Sawaragi, T., Iwai, S. and Sue, V. (1984), "An Intelligent Interface System for the Analysis and Design of Sensing Devices," *Proc. of 1984 International Conf. on Industrial Electronics, Control and Instrumentation*, Tokyo, Japan, 503-508.

Keen, P.G.W. and Scott-Morton, M.S. (1978), "Decision Support Systems: An Organizational Perspective," Addison-Wesley.

Keeney, R.L. and Raiffa H. (1976), "Decisions with Multiple Objectives: Preferences and Value Tradeoffs," John Wiley and Sons, N.Y.

Kelley, H.H. (1972), "Causal Schemata and the Attribution Process," General Learning Press., Morristown, NJ.

Klein, J.H. and Cooper, D.F.(1982), "Cognitive Maps of Decision-Makers in a Complex Game," *Journal of Operational Research Society*, **33**, 63-71.

Kolodner, J.L. (1984), "Towards an Understanding of the Role of Experience in the Evolution from Novice to Expert," Developments in Expert Systems, Academic Press, London, 95-117.

Kolodner, J.L. (1983), "Maintaining Organization in a Dynamic Long-Term Memory, *Cognitive Science*, **7**, 243-280.

Kuipers, B. (1984), "Commonsense Reasoning about Causality: Deriving Behavior from Structure," *Artificial Intelligence*, **24**, 169-203.

Lebowitz, M. (1983), "Generalization from Natural Language Text," *Congnitive Science*, **7**, 1-40.

Lehnert, W.G.(1982), "Plot Units," in Lehneht, W.G.(etal.) *Strategies for Naturall Language Processing*, Lawrence Erlbaum Assoc., Hillsdale, NJ.

Lehnert, W.G. (1981), "Plot Units and Narrative Summarization," *Cognitive Science*, **4**, 293-331.

Lehnert, W.G., Dyer, M.G., Johnson, P.N., Yang, C.J. and Harley, S.(1983), "BORIS-An Experiment in In-Depth Understanding of Narratives," *Artificial Intelligence*, **20**, 15-62.

Lindsay, P.H. and Norman, D.A. (1977), "Human Information Processing," Academic Press Inc., New York.

Methlie, L.B. and Sprague, R.H. (Eds.) (1985), "Knowledge Representation for Decision Support Systems," Elsevier Science Pub., North-Holland.

Michalski, R.S. (1980), "Pattern Recognition as Rule-Guided Inductive Inference," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **PAMI-2**, 349-361.

Michalski, R.S. (1983), "A Theolry and Methodology of Inductive Learning," *Artificial Intelligence*, **20**, 111-161.

Michalski, R.S., and Stepp, R.E. (1983), "Learning from Obsevation: Conceptual Clustering," in Michalski, R.S., Carbonell, J.G. and Mitchell T.M. (Eds.), *Machine Learning: An Artificial Intelligence Approach*, Tioga Pub. Co., Palo Alto, CA., 331-363.

Miller, G.A. (1956), "The Magical Number Seven, Plus Minus two: Some Limits on Our Capacity for Processing Information," *Psychological Review*, 63, 81-97.

Minsky, M. (1975), "A Framework for Representing Knowledge," in Winston, P. (Ed.), *The Psychology of Computer Vision*, McGraw-Hill, New York.

* Nakamura, K., Iwai, S. and Sawaragi, T., (1982a), "A Knowledge Base System for Decision Support Using Cognitive Maps," *Preprints of IFAC Analysis, Design and Evaluation of Man-Machine Systems*, 319-326.

* Nakamura, K., Iwai, S. and Sawaragi, T., (1982b), "Decision Support Using Causation Knowledge Base," *IEEE Transactions on System, Man and Cybernetics*, **SMC-12**, 765-777.

Neisser, U. (1976), "Cognition and Reality," W.H. Freeman and Company.

Norman, D.A. and Rumelhart, D.E. (1976), "Explorations in Cognition," Freeman, San Francisco.

Norman, D.A. (1979), "Descriptins: An Intermaediate Stage in Memory Retrieval," *Cognitive Psychology*, **11**, 107-123.

Osgood, C.E., Saporta, S., and Nunnally, J.C. (1956), "Evaluative Assertion Analysis," *Litera*, **3**, 47-102.

Osherson, D.N. and Smith, E.E. (1981), "On the Adequacy of Prototype Theory as a Theory of Concepts," *Cognition*, **9**, 35-58.

Pereira, F.C.N. and Warren, D.H.D. (1980), "Definite Clause Grammars for Language Analysis," *Artificial Intelligence*, **13**, 231-278.

Quillian, M.R.(1968), "Semantic Memory." in Minsky, M.L. (Ed.), *Semantic*

Information Processing, MIT Press.

Raiffa, H. (1968), "Decision Analysis," Addison-Wesley, Reading, Mass.

Reed, S.K. (1972), "Pattern Recognition and Categorization," *Cognitive Psychology*. 3, 382-407.

Rieger, C. (1975), "Conceptual Memory," in Schank, R. (Ed.), *Conceptual Information Processing*, Amsterdam: North Holland.

Riesbeck, C.K. (1984), "Knowledge Reorganization and Reasoning Style," *Int. J. Man-Machine Studies*, **20**, 45-61.

Riesbeck, C.K. (1978), "Failure-Driven Reminding for Incremental Learning," *Proc. of International Conference on Cybernetics and Society*, Tokyo.

Roberts, R.B., and Goldstein, I.P. (1977), "The FRL Primer," Memo No.408, Artificial Intelligence Laboratory, MIT.

Robey, D. and Taggart, W. (1982), "Human Information Processing in Information and Decision Support Systems," *MIS Quarterly*, **6**, 61-73.

Rosch, E. (1973), "Natural Categories," *Cognitive Psychology*, **4**, 328-350.

Rosch, E. (1978), "Principles of Catergorization," in Rosch, E. and Lloyd, B.B. (Eds.), *Cognition and Categorization*, Lawrence Erlbaum Associates, Pub., Hillsdale, NJ.

Rosch, E. and Mervis, C.B. (1975), "Family Resemblances: Studies in the Internal Structure of Categories," *Cognitive Psychology*, **7**, 573-605.

Rosch, E., Mervis, C.B., Gray. W.D., Johnson, D.M. and Boyes-Braem, P. (1976), "Basic Objects in Natural Categories," *Congnitive Psychololgy*, **8**, 382-439.

Rouse, W.B. (1983), "Models of Human Problem Solving: Detection, Diagnosis, and Compensation for System Failures," *Automatica*, **19**, 613-625.

Rumelhart, D.E. (1975), "Notes on a Schema for Stories," in Bobrow, D.G.

and Collins, A.M.(Eds.), *Representation and Understanding: Studies in Cognitive Science*, Academic Press, New York, 211-236.

Saaty, T.L. (1980), "The Analytic Hierarchy Process," McGraw-Hill, 1980.

Sage, A.P. (1984), "Models for Knowledge Representation and Cognitive Effort in Human Problem Solving," *Preprints of IFAC 84*, Budapest.

* * Sawaragi, T. and Iwai, S. (1987a), "An Expectation-Driven Decision Support by Learning Scenarios from Episodes," *Preprints of 2nd IFSA Congress*, Vol.2, 780-784.

* * Sawaragi, T. and Iwai, S. (1987b), "Cognitive Simulation for Intelligent Decision Support," in Ras, W.R. and Zemankova (eds.), *Methodologies for Intelligent Systems*, Elsevier Science Pub. Co., Inc., New York (Proc. of 2nd International Symposium on Methodologies for Intelligent Systems, Charlotte, USA, 1987), 362-369.

* * Sawaragi, T., Iwai, S., Katai, O. and Sawanoi, A. (1984), "A Measure for Evaluating Structural Ambivalence and its Application to an Interactive DSS," *Preprints of 2nd Knowledge Engineering Symposium held by SICE*, 69-74. (in Japanese)

* * Sawaragi, T., Iwai, S., Katai, O. and Kubo, S. (1985a), "Organizing Multi-Layered Causal Knowledge Structure through Aggregation and Generalization for Decision Support," *Preprints of 3rd Knowledge Engineering Symposium held by SICE*, 7-12. (in Japanese)

* * Sawaragi, T., Iwai, S. and Katai, O. (1985b), "Qualitative Evaluation of Taking Policies in Societal Problems Based on Causation Network and its Application to Knowledge-Based Decision Support System," *Preprints of 1st Fuzzy System Symposium held by IFSA*, 93-98. (in Japanese)

* * Sawaragi, T., Iwai, S. and Katai, O. (1986a), "Studies on Causal Effect and Decisionmaker's Subjective Criteria of Judgment for Causation-Knowledge-Based Decision Support," *Trans. SICE*, **22**, 629-636. (in

Japanese)

* Sawaragi, T., Iwai, S. and Katai, O. (1986b), "An Integration of Qualitative Causal Knowledge for User-Oriented Decision Support," *Control-Theory and Advanced Technology*, **2**, 451-482.

* Sawaragi, T., Iwai, S. and Katai, O.(1986c), "Intelligent DSS Based on Aggregation of Causal Knowledge into Schemata Hierarchy System," *Preprints of 29th Joint Conference of Automatic Control held by SICE, JSME and JAACE*, 409-412. (in Japanese)

* Sawaragi, T., Iwai, S. and Katai, O. (1987a), "Scenario Learning from Episodes and its Application to Decision Support System," *Preprints of 5th Knowledge Engineering Symposium held by SICE*, 93-98. (in Japanese)

* Sawaragi, T., Iwai, S. and Katai, O. (1987b), "A Human-Friendly Interface System for Decision Support Based on Self-Organized Multi-Layered Knowledge Structures," in Sawaragi, Y. (et al.), (Eds.), *Toward Interactive and Intelligent Decision Support Systems*, Springer-Verlag, Berlin, Heidelberg, (Proc. of 7th International Conference on Multiple Criteria Decision Making, Kyoto, Japan, 1986), Vol.2, 30-39.

* Sawaragi, T., Iwai, S. and Katai, O.(1987c), "Studies on Multi-Layered Organization of Causal Knowledge for Grasping Deep Structures Implied in Societal Phenomena," *Trans. SICE*, **23**. 977-984. (in Japanese)

* Sawaragi, T., Iwai, S. and Katai, O.(1987d), "Conceptual Clustering Based on Prototype Theory, *Human Interface News and Report*, **2**, 241-248. (in Japanese)

Schank, R.C. (1972), "Conceptual Dependency: A Theory of Natural Language Understanding," Cognitive Psychology, **3**, 552-631.

Schank, R.C. (1975), "The Sturucture of Episodes in Memory," in Bobrow, D.G. and Collins, A. (Eds.), *Representation and Understanding:*

Studies in Cognitive Science, Academic Press, New York, 237-272.

Schank, R.C. (1975), "Conceptual Infomation Processing," American Elsevier, New York.

Schank, R.C. and Abelson, R.P. (1977), "Scripts, Plans, Goals, and Understanding: An Inquiry into Human Knowledge Structures," Lawrence Erlbaum, Hillsdale, NJ.

Schank, R.C. (1982), "Dynamic Memory, ATheory of Reminding and Learning in Computers and People," Cambridge University Press, New York.

Schank, R.C. and Carbonell, J.G. (1979), "RE: The Gettysburg Address, Representing Social and Political Acts, in Findler (Ed.), *Associative Networks*, Academic Press, New York.

Schank, R.C. and Riesbeck, C.K. (1981), "Inside Computer Understanding; Five Programs Plus Miniatures," Lawrence Erlbaum Associates, Hillsdale, NJ.

Schank, R.C. (1980), "Language and Memory," *Congnitive Science*, **4**, 243-284.

Scott-Morton, M.S. (1971), "Management Decision support Systems: Computer-based Support for Decision Making," Harvard Univ. Press.

Simon, H.A. (1969), "The Science of Artificial," The MIT Press.

Simon, H.A. (1977), "The New Science of Management Decision," Prentice-Hall, Inc.

Simon, H.A. (1979), "Models of Thought," Yale University Press.

Slovic, P., Fischoff, B. and Lichtenstein, S. (1976), "Cognitive Processes and Societal Risk Taking," in Carroll, J.S., and Payne, J.G. (Eds.), *Cognition and Social Behavior*, Lawrence Erlbaum, Hillsdale, NJ., 165-184.

Slovic, P., Fishichhoff, B. and Lichtenstein, S. (1977), "Behavioral Decision Theory," *Ann. Rev. Psychology*, **28**, 1-39.

Sol, H.G. (1982), "Processes and tools for Decision Support," North-Holland

Publishing Company, Amsterdam.

Sprague, R.H. and Carlson, E.D. (1982), "Building Effective Decision Suppor Systems," Prentice-Hall.

Steinbruner, J.D. (1974), "The Cybernetics Theory of Decision: New Dimensions of Political Analysis," Princeton University Press.

Stevens, A., Collins, A. and Goldin, S.E. (1979), "Misconceptions in Student's Understanding," Academic Press Inc., London, 145-156.

Stevens, A., Collins, A. and Beranek, B. (1977), "The Goal Structure of a Socratic Tutor," *Proc. of Annual Conf. of ACM*, 256-263.

Szaniawski, K. (1980), "Philosophuy of Decision Making," *Acta Psychologica*, **45**, 327-341.

Thorndyke, P.W. and Hayes-Roth, B. (1979), "The Use of Schemata in the Acquisition and Transfer of Knowledge," *Cognitive Psychology*, **11**, 82-106.

Tuggle F.D. and Barron F.H. (1980), "On the Validation of Descriptive Models of Decision Making," *Act Psychologica*, **45**, 197-210.

Tversky, A. and Kahneman, D. (1973), "A Heuristic for Judging Frequency and Probability," *Congnitive Psychololgy*, **5**, 207-232.

von Neumann, J. and Morgenstern, O. (1947), "Theory of Games and Economic Behavior," Princeton Univ. Press, Princeton, NJ.

von Winterfeldt, P. (1980), "Structuring Decision Problems for Decision Analysis," *Acta Psychologica*, **45**, 71-93.

Warfield, J.N. (1976), "Societal Systems ··· Planning, Policy and Complexity," John Wiley and Sons.

Winston, P.H. (1980), "Learning and Reasoning by Analogy," *Communications of the ACM*, **23**, 689-703.

Winston, P.H. (1982), "Learning New Principles from Precedents and

Exercises," *Artificial Intelligence*, **19**, 321-350.